Anatomy of Threats to The Internet of Things

Imran Makhdoom[1], Mehran Abolhasan[2], Justin Lipman[3], Ren Ping Liu[4], Wei Ni[5]
[1,2,3,4] University of Technology, Sydney, Australia, [4]Data61-CSIRO
imran.makhdoom@student.uts.edu.au[1], mehran.abolhasan@uts.edu.au[2],
Justin.Lipman@uts.edu.au[3], RenPing.Liu@uts.edu.au[4], Wei.Ni@data61.csiro.au[4]

***Abstract*—The world is resorting to the Internet of Things (IoT) for ease of control and monitoring of smart devices. The ubiquitous use of IoT ranges from Industrial Control Systems (ICS) to e-Health, e-Commerce, smart cities, supply chain management, smart cars, Cyber Physical Systems (CPS) and a lot more. Such reliance on IoT is resulting in a significant amount of data to be generated, collected, processed and analyzed. The big data analytics is no doubt beneficial for business development. However, at the same time, numerous threats to the availability and privacy of the user data, message and device integrity, the vulnerability of IoT devices to malware attacks and the risk of physical compromise of devices pose a significant danger to the sustenance of IoT. This paper thus endeavors to highlight most of the known threats at various layers of the IoT architecture with a focus on the anatomy of malware attacks. We present a detailed attack methodology adopted by some of the most successful malware attacks on IoT including ICS and CPS. We also deduce an attack strategy of a Distributed Denial of Service attack through IoT botnet followed by requisite security measures. In the end, we propose a composite guideline for the development of an IoT security framework based on industry best practices and also highlight lessons learned, pitfalls and the open research challenges.**

***Index Terms*—Threats to the IoT, Internet of Things, malware attacks on the Internet of Things, attack methodology, security and privacy, IoT security framework, security guidelines.**

I. Introduction

Millions of embedded devices are being used today in safety and security critical applications such as Industrial Control Systems (ICS), Vehicle Ad-Hoc Networks (VANET), disaster management and critical infrastructure [1]. A massive number of these devices have been interconnected to each other and further connected to the internet to form an Internet of Things (IoT). IoT based services have seen an exponential economic growth in last five years especially in telehealth and manufacturing applications and are expected to create about $1.1- $2.5 Trillion contribution in the global economy by 2020 [2]. It is estimated that by 2020, the number of IoT connected devices will exceed to 30 billion from 9.9 million in 2013 [3] and M2M (Machine-to-Machine) traffic flows are also expected to constitute up to 45% of the whole internet traffic [4]. However, due to interconnection with the internet, IoT devices are vulnerable to various attacks [1, 5, 6, 7, 8, 9, 10]. Moreover, it is believed that IoT devices are being manufactured rapidly without giving much attention to security challenges and the requisite threats [11].

According to [12], more than 85% of enterprises around the world will be turning to IoT devices in one form or the other, and 90% of these organizations are not sure about the security of their IoT devices. Similarly, Joseph Steinberg in [13] has listed many appliances that can spy on people in their own homes. A recent study carried out by HP [14] also revealed that 70% of the devices connected to the internet are vulnerable to numerous attacks. Moreover, development of smart cars is also on the rise in the world, in which vehicle on-board computer systems are connected to the internet thus making them vulnerable to Cyber-attacks [7]. In addition, the legacy industrial systems such as manufacturing, energy, transportation, chemical, water and sewage control systems (connected by IoT to achieve better monitoring, control, and conditional maintenance) have greater security risks [15]. Attacks on industrial systems are not just a threat instead it is a reality, as two Russian security researchers found vulnerabilities in more than 60,000 internet connected control systems that could be exploited to take full control of the compromised systems running energy, chemical, and transportation applications

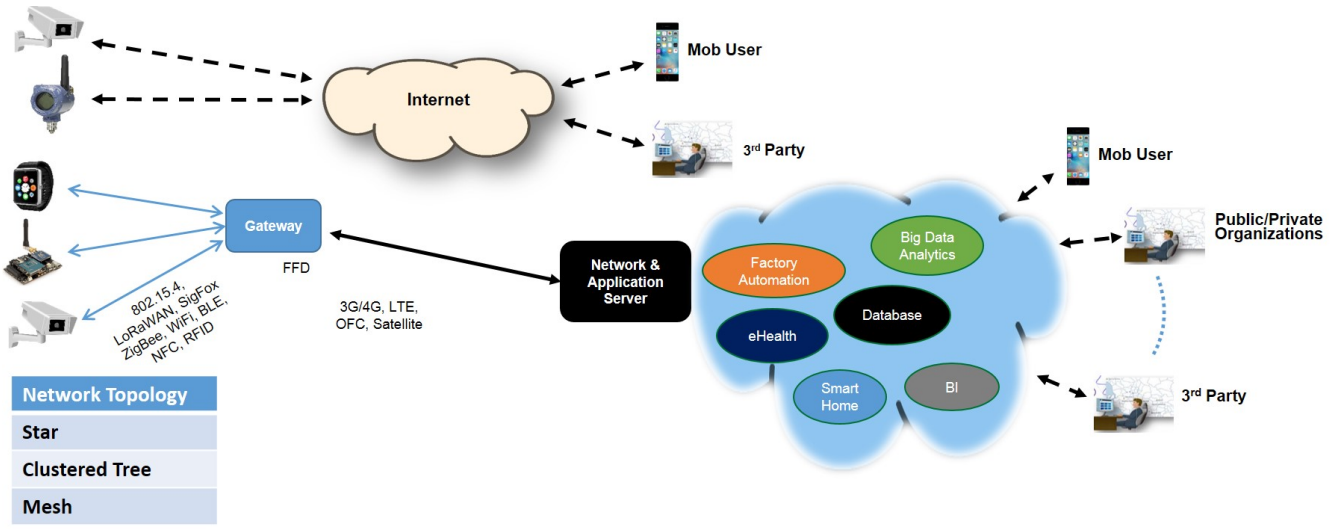

Fig. 1: Generalized IoT Architecture

[16]. Furthermore, it is expected that by the end of 2020, more than 25% of corporate attacks would be because of compromised IoT devices [17]. Similarly, the successful launch of sophisticated cyber-attacks like Mirai [18], Ransomware [19], Shamoon-2 [20] and DuQu-2 [21] on ICS and other critical infrastructure in recent past have rendered existing IoT protocols ineffective.

*A. **Related Work***

Till date many reviews and survey papers [8, 10, 22, 23, 24, 25, 26] have been published to highlight the security issues of IoT. However, they do not cover the full spectrum of IoT security. A detailed comparison of existing work is shown in Table-I. Most of the current work focuses on few aspects and leaves the rest. For instance, [8] refers to limited security issues at different IoT layers and discusses all theoretical/non-industrial security methods without defining an overall security model. Similarly, [10] mostly enumerates the DoS (Denial of Service) attacks on various layers of WSN (Wireless Sensor Network) and some security vulnerabilities in RFID technology. It does not give examples of such attacks illustrating the vulnerabilities exploited and also lacks recommended security measures to protect against mentioned attacks. Whereas, [22] highlights some generalized IoT security gaps concerning lack of standardization and regulations by discussing pros and cons of some existing security frameworks such as COBIT, ISO/IEC 27002:2005. It proposes an integrated security framework with generalized recommendations on hardware and protocol security with an urge to develop IoT specific security standards.

Authors in [23], also briefly discuss the security and privacy issues in IoT with focus on some open problems. The paper broadly covers some of the generalized security and privacy threats including internal and external attacks, DoS attacks, physical attacks and attacks on privacy. Authors also highlight some of the security and privacy challenges to IoT such as user privacy, data protection/authentication, identity/trust management, authorization and access control. Whereas, [24] covers the security and open research issues related to IoT communication protocols only. Similarly, [25] briefly highlights some security and privacy issues of five smart-home devices and proposes an SDN-based network level security mechanism that monitor and control network operations of each IoT device.

In another notable work [26], authors present an IoT security architecture comprising three layers, i.e., perception, transport and application layer. The paper comprehensively covers security issues of IoT with a focus on RFID and WSN. They also discuss access network technologies including WiFi and 3G. Although authors have amply covered some security issues related to IoT, yet there is a roam of improvement by including examples of practical attacks/vulnerabilities in IoT such as smart-home and wearable IoT devices. There is a further requirement of adding a comprehensive security framework for IoT. Resultantly, there is a need of a comprehensive illustration of practical threats to IoT and formulation of a set of security guidelines that should cater for varying

TABLE I:
Comparison of Existing Surveys

Existing Survey	Consolidated Introduction to IoT	Illustration of generalized and threats at IoT layers	Threats to IoT Communication Protocols	Examples of real-world attacks	Security issues of Cloud-based IoT and Fog computing	Malware Threat	IoT Botnets	Defense-in-Depth security measures	Summary of threats to IoT and associated vulnerabilities	Open research issues
[8]	X	Limited security issues at IoT layers	X	X	X	X	X	Theoretical security solutions	X	X
[10]	X DoS attacks in WSN and some security issues in RFID	X	X	X	X	X	X	X	X	X
[22]	X	Generalized security gaps concerning IoT standardization	X	X	X	X	X	•Pros and Cons of existing security frameworks, e.g., COBIT, ISO/IEC 27002:2005 •Generalized recommendations for hardware and protocol security	X	X
[23]	X	•Broadly covers generalized security and privacy threats •Internal and external attacks •Physical attacks and attacks on user privacy	X	X	X	X	Simple DoS attacks	X	X	X
[24]	X	X	√	X	X	X	X	X	X	IoT communication protocols only
[25]	X	Security and privacy issues in some smart home devices	X	X	X	X	X	Proposes an SDN-based network level security mechanism	X	X
[26]	X	Security issues in WSN and RFID	X	X	X	X	X	Proposes an IoT security architecture comprising perception, transport and application layer	X	X

standards of IoT devices and recommend a common framework for end-to-end IoT security [17].

Contributions of the paper. To cover the gaps in current literature (as shown in Table-1), the major contribution of this paper is to present an "All in one package" that comprehensively covers most of the aspects of IoT security. The paper develops logically by first introducing a generalized IoT architecture and a detailed IoT protocol stack showing technologies, protocols and functionalities at various layers of IoT. It amply covers a range of generalized as well as specific threats at different layers of IoT with examples of such attacks on IoT systems/devices at most of the places. We also present a consolidated list of threats to IoT along with the vulnerabilities that can be exploited to convert these threats into successful attacks. Another aspect that differs this paper from its predecessors is its due diligence on malware attacks and their attack methodology. We also deduce an attack strategy of a Distributed Denial of Service (DDoS) attack through IoT botnet followed by necessary security measures. This paper also presents a comprehensive set of security guidelines based on industry best practices that can help IoT standardization bodies to design minimum security standards based on types of IoT applications. Finally, some open research challenges related to IoT security are discussed.

B. *Paper Organization*

The rest of the paper is organized as follows: Section-2 presents a detailed description of threats to IoT. Attack methodology of some of the most successful malware attacks is described in Section-3, while the gap analysis, attack strategy of a DDoS attack on IoT devices, and guidelines for the security framework are discussed in Section-4. Summary, lessons learnt and pitfalls are illustrated in Section-5. In Section-6, we present some open research challenges, and finally, the paper is concluded with some description of the future work in Section-7.

II. Threats to The IoT

This section presents a detailed description of some generalized and various specific threats to different layers of IoT architecture. However, before we do the threat modeling, it is essential to explain the IoT architecture and some important terms that would be used frequently in the later text. Firstly, IoT systems and IoT ecosystem would be encountered often. Where, IoT system refers to a typical IoT application like smart-home, smart-grid, smart-vehicle, smart-watch, etc., and IoT ecosystem points to the IoT (with all its applications) as a whole. Secondly, IoT architecture concerns the way different objects such as sensors, actuators, gateways, network and application servers are arranged and communicate with each other.

A. *IoT Architecture*

Currently, there is a lack of consistency and standardization in IoT solutions across the globe due to which there are issues related to interoperability, compatibility, and manageability [27]. Likewise, non-uniformity in the presentation of IoT Architecture and layered protocol stack was observed in the literature review [8, 24, 28, 29, 30, 31, 32, 33, 34, 35, 36]. Such as, [8] presents IoT layers showing the meagre detail of functionality and the protocols. Similarly, [24] just focuses on communication protocols at various IoT layers. Whereas, [28] displays a table of elements and technologies that together form an IoT. Therefore, it is believed that due to this non-standardization, the world has not yet been able to agree on a single IoT reference model [28]. To reduce this non-uniformity, we present a consolidated generalized IoT architecture and a layered IoT protocol stack shown in Figure-1 and Figure-2 respectively. An IoT ecosystem may comprise different types of devices, which can be deployed in any of the following topologies, i.e., star, clustered tree, and mesh. "Things" are usually connected to a gateway device using various IoT communication protocols such as 802.15.4, LoRaWAN, SigFox, ZigBee, WiFi, Bluetooth Low Energy (BLE), Near Field Communication and Radio Frequency Identification (RFID). The gateway device is connected to an application or a network server via 3G/4G, LTE (Long-Term Evolution), Optical Fiber Cable (OFC), satellite link, etc. The network/application servers (can be located in the cloud) provide different data analytic services to its users and third parties including government and private organizations. The processed data is turned into useful information in the form of health statistics, smart home autonomous services, business intelligence, industrial automation, environmental monitoring, livable urban communities and smart city sharing services.

As far as IoT protocol stack is concerned, the first layer is the physical/perception layer that consists of sensors, actuators, computational hardware, identification and addressing of the things. As the name suggests, its purpose

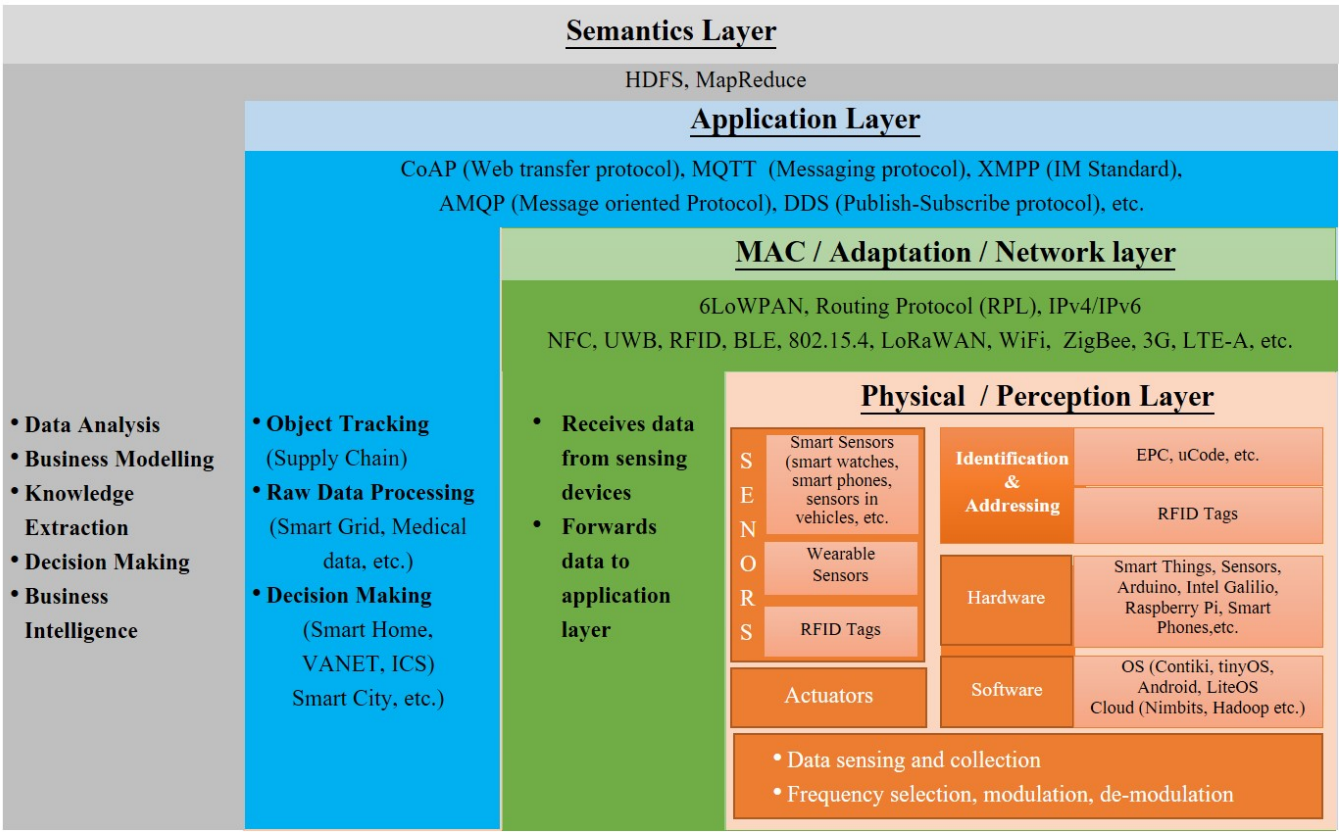

Fig. 2: IoT Protocol Stack

is to perceive the data from the environment. All the data collection and data sensing is done at this layer [37]. Some other functions of physical layer include frequency selection, modulation-demodulation, encryption-decryption, transmission and reception of data. The challenges faced by this layer are energy consumption, security, and interoperability [27]. The second layer is the MAC (Medium Access Control)/Adaptation/Network layer, which is responsible for receiving data from sensing devices and then forward it to the application layer for processing, analytics, and smart services. The network layer also faces specific issues concerning scalability, network availability, power consumption and security [27].

The third layer is the application/services layer which provides smart services to the customers and also feeds processed/aggregated data to the semantics layer. The challenges being faced at this layer are related to handling, storage, and processing of data received from the sensors, security/privacy of user information and conformity to industrial/government regulations. E.g., Health Insurance Portability Accountability Act (HIPAA) in the United States and Personal Information Protection and Electronic Documents Act (PIPEDA) in Canada, protect the users' rights concerning their health and personal information. The fourth and the last layer is semantics which can also be termed as a business management layer as it manages all the activities of an IoT system. It implies the use of cognitive technologies to provide certain high-end services such as data analysis, business intelligence, strategic decision-making and business modelling.

Although, by now we are clear about what IoT is, however there are many areas in which IoT is different than tradition networks (including LANs and internet), which are being discussed in succeeding paras.

B. IoT vs Traditional Networks

Before discussing IoT threats, it is important to understand the differences between IoT and traditional networks, as these differences influence the development of requisite security and privacy solutions for IoT systems. The significant difference between conventional networks and IoT is the level of the resourcefulness of end devices [26]. IoT usually comprises resource constraint embedded devices such as RFID and sensor nodes. These devices have low memory, low computing power, small disk

space and require low power consumption. Whereas, the traditional internet is composed of computers, servers and smartphones that have plentiful resources. Hence, the traditional networks can be supported by complex and multi-factor security protocols without any resource consideration. In contrast to this, IoT systems require lightweight security algorithms that maintain a balance between security and resource consumption such as battery life.

IoT devices mostly connect to the internet or gateway devices through slower and less secure wireless communication media such as 802.15.4, 802.11a/b/g/n/p, LoRa, ZigBee, NB-IoT and SigFox. Resultantly, IoT systems are prone to data leakage and other privacy issues. Whereas, in the traditional internet, end devices communicate through more secure and faster wired/wireless media such as fiber optics, DSL/ADSL, WiFi, 4G and LTE. Another difference is that traditional network devices have almost the same OS and data format, but in the case of IoT because of application-specific functionality and lack of OS, there are different data contents and formats. Hence, because of this diversity, it is difficult to develop a standard security protocol that fits all types of IoT devices and systems. As a result, a wide range of IoT threats are still at loose and threaten the security and privacy of the users.

If we look at the security design, traditional networks are secured by a blend of static network perimeter defense based on firewalls, IDS/IPS and the end devices are secured by host-based approaches such as anti-virus and security/software patches. Whereas, the host-based security approach cannot be applied to the resource constraint IoT devices. Similarly, traditional perimeter defense mechanism cannot secure IoT devices, since these devices are deployed deep in the network. Hence, it is concluded by authors in [38] that IoT devices cannot be protected only by host-based solutions.

C. *Generalized Threats*

It is estimated that with the rise in number of things connected to IoT systems to swarming billions of devices by 2020, the potential vulnerabilities will also increase [22]. Hence, the increase in vulnerabilities due to non-standardization of IoT technologies may give rise to security incidents in IoT systems. Some of the most common security issues in IoT are highlighted in succeeding sections.

1) ***Security and Privacy Issues****:* During a security audit conducted by [39], numerous smart devices were checked for security breaches. As per findings of the security audit, almost 90% of these devices gather personal information about the users in some form or the other. This unauthorized storage of information is vulnerable to data security, privacy and integrity attacks. Researchers in [9] and [22] have also rendered security and privacy issues a threat to data confidentiality and user privacy. Moreover, lack of reliable authentication mechanism in IoT devices is also a contributing factor in weak IoT security [10]. Additionally, the lack of data encryption and network access control measures enable an attacker to pose a real threat to user privacy as a result of eavesdropping and traffic analysis [40].

2) ***Threats to eHealth IoT Devices****:* Biomedical Sensor Network (BSN) is a specialized case of WSN in which sensors are used to monitor patients' health and also facilitate chronic disease self-care [41]. BSN has dynamic network topology due to mobile nodes, power constraints and low bandwidth IoT communication protocols. Therefore, BSN is vulnerable to numerous attacks including DoS, eavesdropping, masquerading, and unauthorized disclosure of personal health information. A successful attack can be life-threatening, and can also cause loss of data, misuse of access, loss of personal information, manipulation of data and even in some cases non-availability of critical health services.

3) ***Device Integrity****:* The deployment and successful operation of IoT in critical infrastructures like smart grids, healthcare, intelligent traffic systems, smart vehicles and smart homes are highly dependent on the reliability of devices and the data transmitted between these devices [8]. However, IoT end devices mostly operate in a trustless environment without any physical security. Hence, these devices are subject to physical attacks including invasive hardware attacks, side-channel attacks, and reverse-engineering attacks [42]. In addition, cyber-attacks incorporating compromised IoT devices as bots such as Mirai DDoS Attack, are a significant threat to corporate IoT [43].

4) ***Software/Code Integrity****:* Software integrity including the integrity of the operating system, applications, and configurations of IoT devices, is a key element to guarantee security and privacy of the “Things”. Recently a practical manifestation of such an attack was experienced by the world, named “Mirai” [44]. This attack created a botnet by hacking into thousands of IoT devices includ-

ing CCTV cameras and DVRs, by exploiting a firmware weakness and then directed these devices to launch a DDoS attack on a DNS (Domain Name System) service provider named DYN.

It is believed that the lack of anti-virus/malware detection mechanism in IoT leads to attacks on the integrity of the code/software of an end device [8, 9]. The mobile applications are another source of malware in smart devices that further corrupt the computer networks through infected emails, documents, and direct connection. In 2016, approximately one million Google accounts were hacked due to an Android malware called "Gooligan". The malware propagated through eightysix seemingly legitimate applications [17]. Therefore, IoT devices need to be protected against malware attacks such as Trojans, viruses, and other runtime attacks [9].

5) ***Issues Concerning Communication Protocols****:* Further challenges in security design of IoT/CPS arise from the fact that most of the current wireless communication protocols adhere to the OSI layered protocol architecture and the physical layer encryption is not complemented with additional security mechanisms in the upper layers of the communication [45]. A MITM (Man-in-the-Middle) attack launched by spoofing the address resolution protocol (ARP) at MAC layer is an example of such a security breach. Moreover, researchers in [46] have identified that cross-layer and hybrid security issues are open research challenges in wireless communication. These issues can be easily extended toward IoT and CPS. Same has been demonstrated through various security breaches such as maliciously gaining unauthorized access to a Mitsubishi vehicle through a brute-force hack of the pre-shared WiFi key, exfiltration of private/sensitive data from a computer through a covert FM channel [47], and hacking of wireless controlled implantable medical device [48].

Similarly, cellular technologies such as UMTS, GSM, and LTE also suffer from specific security issues [49]. Due to open implementation of radio baseband stacks, the mobile networks have an added threat of hacking and cyber-attacks. Moreover, GSM and UMTS networks are vulnerable to "IMSI Catching" by an active attacker. In addition, there is a time delay in setting security contexts while a UE (User Equipment) is connected to the base station. Such a delay may prove fatal for delay-sensitive applications, e.g., autonomous cars, smart medical instruments, etc. Mobile networks are also vulnerable to DoS attacks launched by mobile bots [49]. The mobile bots may attack MME (Mobile Management Entity) and HSS (Home Subscriber Server). Correspondingly, radio interface jamming is the DoS attack specific to wireless communication. A smart jamming attack can be launched against 3GPP (3rd Generation Partnership Project) specified mobile networks by using mobile botnets, in which control channels essential for the overall operation of the radio interface can be selectively blocked. DoS attacks are even a threat to 5G networks.

Furthermore, the short-range wireless technologies like Bluetooth and Zigbee are not suitable for applications that require long communication range with low bandwidth. Although, cellular technology does provide long coverage for M2M communication, but require more power [50]. Therefore, since 2015, LPWAN (Low Power Wide Area Network) technology is considered to be a suitable technology for the applications that require wide area coverage, low energy consumption, QoS (Quality of Service), low data transmission rate, low latency and low costs [50, 51].

Koushanfar et al. also illustrate that communication protocols are subject to protocol attacks, including MITM and DoS attacks [52]. A manifestation of one of the DoS attacks on the wireless communication protocol 802.11b is presented in [53]. The author highlights the vulnerability in the exchange of disassociation message between the client and the station. It is identified that the message is sent without any authentication. Hence, it enables an attacker to initiate a disassociation message on behalf of other users to stop them from connecting to the network. Correspondingly, this DoS can result in a severe availability issue in case of a CPS/IoT system [54]. It can further be deduced that almost all the communication protocols such as 802.15.4, Zigbee and LoRaWAN provide conventional cryptographic security assurances such as confidentiality, data integrity, data authenticity, replay protection and non-repudiation [24, 30]. However, the cryptographic security embedded in communication protocols is not meant to protect against node compromise and malware attacks.

There is another upcoming communication technology, being developed by IEEE 802.1 TSN (Time Sensitive Networks) TG (Task Group) for applications requiring Ultra-Low Latency (ULL). TSN promises a secure end-to-end network connection between a sender and receiver node through a time-sensitive capable network [55]. Similarly,

IETF (Internet Engineering Task Force) is also working on DetNet (Deterministic Networks) to interconnect the isolated OT (Operational Technology), i.e., CPS with IT networks [56]. However, such an interconnection will expose the CPS to various internal and external attacks. Moreover, being a work in progress, security aspects require due consideration to mitigate the internal and external threats ranging from detNet flow modification to path manipulation and attacks on Time Synchronized Mechanisms.

Coming over to the core network communication media, mostly OFC interconnects multiple corporate data centres or an ISP with the internet gateway. An optical fiber channel may directly impact an IoT system, e.g., a smart home gateway device is connected to an ISP through a Fiber-To-The-Home (FTTH) connection in order to provide internet-based remote access to various services to the owner of the house and same connection can be used by the vendor for maintenance/remote monitoring of the system. Optical channels are vulnerable to eavesdropping, jamming and attacks to the availability [57]. An attacker can eavesdrop on classified/private data by tapping into an optical fiber for unencrypted channels [58] or by cracking the encryption keys that are isolated from the payload and are transferred over the Network Management System (NMS) [59]. Whereas, jamming attacks can be launched by introducing in-band and out-of-band cross-talk [60], and by exploiting vulnerabilities of the alien wavelengths [61]. Some other factors that may degrade an optical channel by launching signal insertion attacks include Mixed Line Rate (MLR) networks, On-Off-Keying (OOK) amplitude modulation and Cross-Polarization Modulation (XPolM) [62].

6) ***Hardware Vulnerabilities*** *:* IoT devices are being commercially developed with more emphasis on device functionality rather than security. Hence, security features are often added in an ad-hoc manner. Therefore, commercial IoT devices have residual hardware vulnerabilities such as open physical interfaces and boot process susceptibilities which can be remotely exploited [63]. Whereas, the reliable and safe operation of IoT systems depends on the integrity of the underlying devices, in particular, the integrity of their code and data against malicious modifications [64].

7) ***DoS Attacks****:* Due to constraint resources such as low memory, low computation power and low battery consumption, IoT devices are vulnerable to resource exhaustion attacks [23]. These attacks include jamming of communication channels, extensive unauthorized or malicious utilization of critical IoT resources such as bandwidth, memory, CPU time, disk space and change of node configuration. All of these attacks will most likely affect the operational functionality of IoT devices and non-availability of their services to the respective users.

8) ***DDoS Attacks****:* The analysis of past cyber incidents infer that the vulnerabilities of IoT devices make them an ideal platform to launch DDoS attacks. It has also been disclosed by [65] that 96 percent of the devices involved in DDoS attacks were IoT devices. Whereas, 3 percent were home routers and 1 percent were compromised Linux Servers.

9) ***Security Challenges Specific to WSN****:* Chen et al. in [66], have classified threats unique to WSN in following categories: interruption, interception, modification, and fabrication attacks. Moreover, unauthorized insertion of malicious messages in the network has also been highlighted by [29]. Authors in [26] point out that due to wireless communication media, the process of information collection/sharing can be subjected to eavesdropping, malicious routing and message tampering.

10) ***Security Issues of RFID and Bluetooth Devices****:* Due to lack of physical protection and wireless nature of RFID communication, RFID tag data is vulnerable to confidentiality and integrity attacks [29]. Some other security issues include lack of uniform coding, conflict collision, privacy protection and trust management between RFID tag and the reader and between reader and the base station [26]. Similarly, use of unpatched or old version Bluetooth devices can cause connectivity to unauthorized/malicious devices thus exposing private or security-critical data [29].

11) ***User Unawareness****:* Users are one of the most common attack vectors. Due to lack of security training and awareness, employees are vulnerable to social engineering, phishing, spear-phishing and accidental security breaches. Hence, they unwittingly download malicious codes by clicking infected links in the emails. In addition, sharing of sensitive data over public networks through mobile devices is another cause of security breaches. It is therefore estimated that with the increase in smartphone users, almost one-third of the mobile devices are at high risk of exposing official data [17].

D. ***Threats at Different Layers of IoT Architecture***

Table-II shows a list of numerous threats at various layers of IoT architecture and the vulnerabilities that can

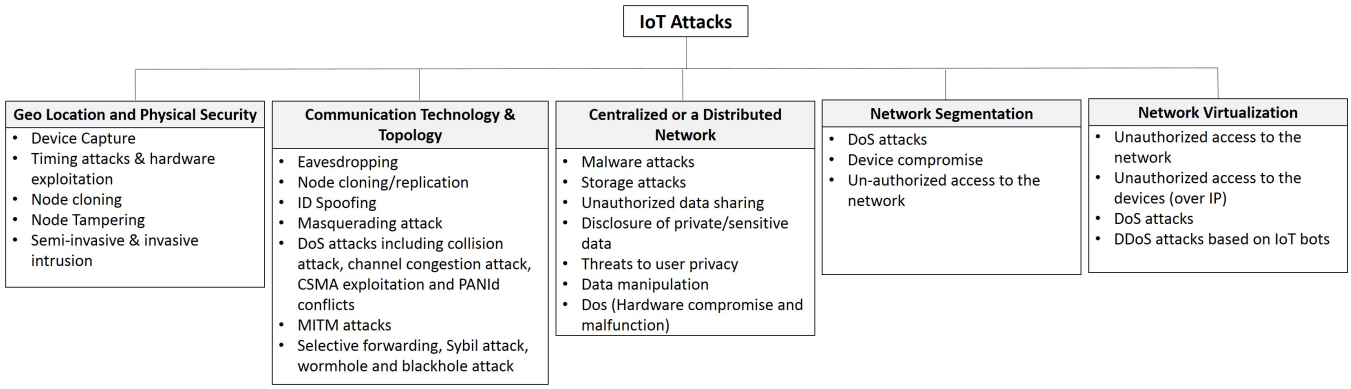

Fig. 3: Classification of IoT Attacks based on their Impact on Deployment

be exploited to convert such risks into successful attacks. Moreover, these attacks have also been classified based upon their impact on IoT node deployment and network architecture. As shown in Figure-3, the IoT attacks affect the geographical (geo) placement/location and level of physical security of IoT devices as per the sensitivity of data and the critical infrastructure. Similarly, selection of IoT communication protocol and network topology is also derived by the threat environment and the requirement of requisite security measures. E.g., if there is a threat of jamming of wireless channels by the attacker, the use of frequency hopping or a spread spectrum technology would be an appropriate response. Similarly, the decision on the network control by a single entity or a distributed control, and other network security paradigms such as the need of network segmentation and network virtualization for better neutralization and mitigation of IoT attacks are also derived by the extent and types of IoT attacks. The detailed description of these threats at different layers of IoT architecture is presented in the succeeding sections.

1) ***Physical/Perception Layer****:* Some of the significant threats at physical/perception layer include:

Eavesdropping on Wireless Communication. Attackers can install devices similar to end nodes in an IoT system to sniff wireless traffic and extract some valuable information about users.

Loss of Power. A Battery drainage attack in which a node is bombarded with a large no of legal requests thus preventing it from going to sleep or energy saving mode.

Hardware Failure. IoT devices installed in ehealth, Intelligent Transport Systems (ITS), smart cities and smart grids can be termed as the lifeline to the users. Hardware failure due to a manufacturing fault or as a result of a cyber-attack may lead to substantial damage to the system and physical impairment to the users [8]. In such an endeavour, researchers from security consultancy Rapid-7 [67] discovered that seven commercially available smart devices are vulnerable to cyber-attacks. These devices include the Philips In.Sight wireless baby monitor, iBaby Monitor M3S/M6, Summer Infant Baby Zoom, TrendNet WiFi Baby Cam, Lens Peek-a-View and a Gynoii device.

In some cases, attacks were as simple as guessing or switching out sections of web addresses/URLs. In the particular case of iBaby M6, it was possible to guess the serial number of the device, the camera type, and a user ID. These parameters were then used in the web login URL to execute an authentication bypass access to the device. In a similar attack, the researchers were able to initiate video and audio streams in a Philips camera. In general, there was no blacklisting or whitelisting of IP addresses to control access to these URLs. The researchers were also able to register a new user account for the Summer Baby Zoom Camera by manipulating the URL related to Summer Baby WiFi Monitor and Internet Viewing System without any disclosure/alarm to the legitimate users.

Malicious Data Injection by Forged Devices. Any determined malicious attacker can introduce a forged device in an IoT system to eavesdrop on the radio traffic, inject fabricated messages or flood the radio channels with fake messages to render the system unavailable to the legitimate users [68].

Sybil Attack. In this attack, a malicious node may present multiple identities by impersonating other nodes or by generating new fake identities. In the worst case scenario, multiple identities may be generated using a

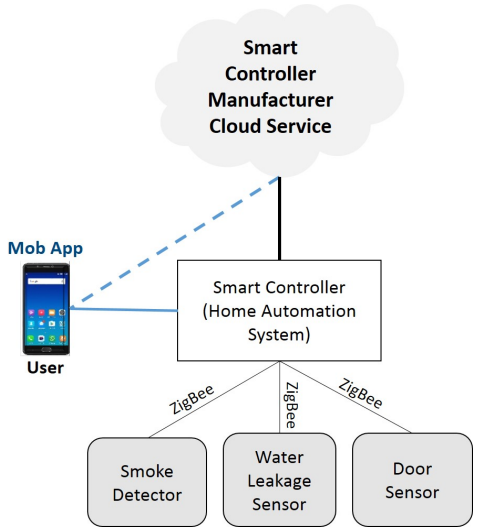

Fig. 4: Home Automation Device Setup

single physical device [69]. The attacker may present all the Sybil Identities simultaneously or one by one at different instances. A Sybil Attack may affect the outcome of a voting-based fault tolerance system or a routing protocol.

Disclosure of Critical Information. A malicious attacker say a smart thief continually monitors the wireless sensors traffic of a smart house. Even if the wireless data is encrypted, the reduced data traffic may infer critical information to the attacker that the house is empty. Therefore, he can plan a robbery.

Side-Channel Attacks. These attacks are based on side-channel information about the encryption device. Such information is other than the plaintext or ciphertext messages, i.e., data about processing time or power consumption of the device in encrypting/decrypting various messages and during the computation of different security protocols like Diffie Hellman (DH) key exchange and Digital Signature Standard (DSS) protocols [70].

Device Compromise. In a practical manifestation of such an attack, researchers in [71] compromised a smart controller of a house (device setup is shown in Figure-4) automation system through an open UART interface. The complete attack sequence is also shown in the Figure-5. Once the researchers gained access to the device, they were able to view the start-up sequence. They modified the boot parameters and gained low-level access to the device. They also brute forced the root password and launched network layer attacks such as port scanning and network traffic analysis. In another vulnerability assessment, the researchers were able to modify the (ID) identity of a smart meter by compromising the device through a JTAG (Joint Test Action Group) interface. They re-enabled write access to an EEPROM (Electrically Erasable Programmable Read-only Memory) that stored the device ID. As a result of such an attack, the spoofed device ID can be used to feed wrong power consumption data to the smart meter reader.

Similarly, owing to the boot process vulnerabilities, the compromise of boot sequence not only facilitates the attackers in attacking other high-level layers but also in taking control of the device. In an experimental setting in [72], a similar attack was successfully executed on Google Nest Learning Thermostat and Nike+ Fuelband SE fitness tracker. The researchers exploited vulnerabilities in the boot process of the Nest Thermostat OS and also some weaknesses in the physical design. The devices were compromised despite the availability of default security features including WPA-2 personal security on WiFi interface, TLS (Transport Layer Security) 1.2 for transmission of any log related data, access to Nest Cloud using OAuth authentication tokens and use of PKCS-7 certificates to ensure authentication and integrity of update images.

Timing Attacks and Hardware Exploitation. Debugging ports (UART (Universal Asynchronous Receiver-Transmitter), JTAG, etc.) left open by the manufacturers make the system vulnerable to timing attacks and re-flashing of external memory [1]. E.g., a weakness in Xbox 360 allows the system to be downgraded to a vulnerable

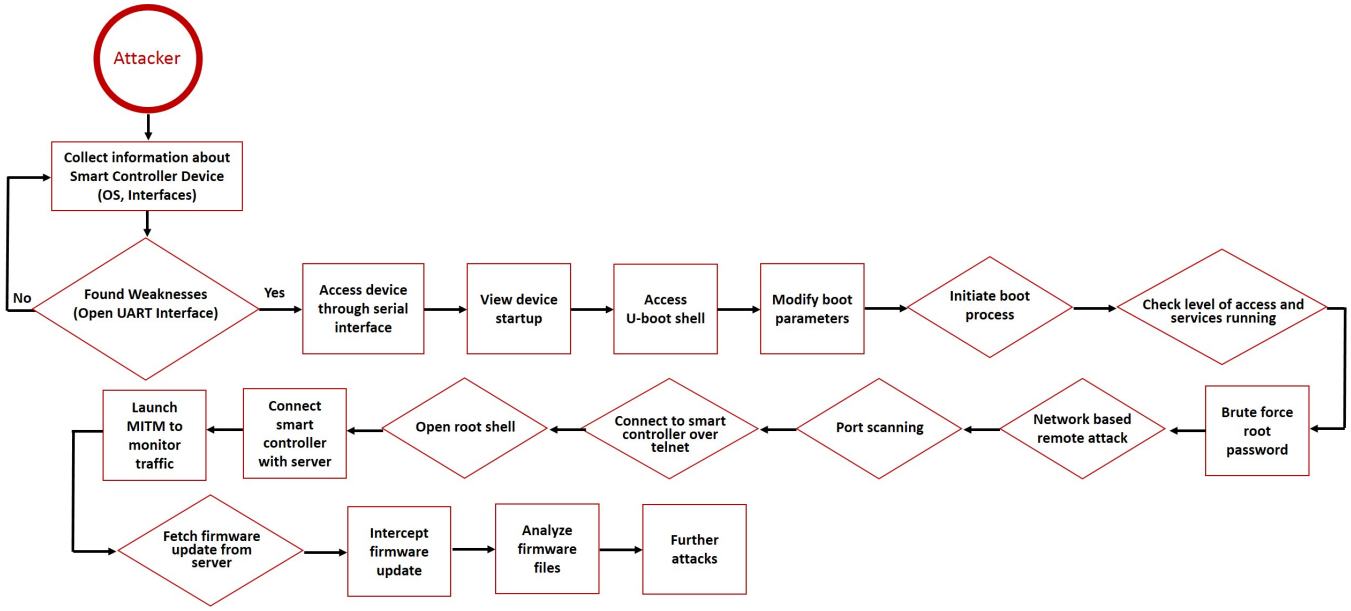

Fig. 5: Attack Sequence of Compromising a Smart Home Controller Through an Open Interface

kernel version through a timing attack [73].

Node Cloning. Due to lack of standardization of IoT device design, mostly the IoT devices such as sensor nodes and CCTV cameras are developed without any hardware tamper-proofing. Therefore, these devices can easily be forged and replicated for unauthorized purposes. This phenomenon is also known as node cloning [74]. It can happen in any of the two phases, i.e., manufacturing phase, as well as during the operational phase. In the former case, an internal attacker can substitute an original device with a similar, pre-programmed thing for unauthorized purposes. Whereas, during the operational phase, a node can be captured and cloned. Node capture could further lead to extraction of security parameters and firmware replacement attacks.

Invasive/Semi-invasive Intrusions. Semi-invasive and invasive intrusions are a serious threat to smart devices, as trusted boot sequence relies on trusted on-chip assets. Since long, encryption/decryption keys and other sensitive information stored on-chip is considered secure. However, today the invasive methods can reveal valuable assets stored on the chip and may compromise any protocol utilizing the secret information. In such an endeavor [75], the researchers were able to extract the stored AES (Advanced Encryption Standard) Key from the internal memory of Actel ProASIC3 FPGA, by launching “Bumping Attacks”.

Change of Configuration/Firmware-Version. Improper implementation of encryption and hash functions threaten the security of the underlying system. E.g., even if a system is secured with robust authentication mechanism such as X.509 certificate-based TLS, unless the credentials are securely stored they can be subject to malicious attacks. Researchers in [76] were able to downgrade the firmware of Sony Play Station-3, by exploiting weak cryptographic implementations.

Unauthorized Access to The Devices. Use of default passwords by the users and hardcoded username and passwords by the manufacturers is a major security vulnerability nowadays. For instance, the iBaby M3S wireless monitor is shipped with a hardcoded username and a password of “admin”. Whereas, the hardcoded credentials can only be fixed by a firmware update from the manufacturer [67]. Moreover, the channels that are left open by the manufacturers for debugging or OTA (Over The Air) firmware update, are not always secure. Hence, developers may leave some open APIs (Application Programming Interface) for executing various commands at a later time. The attackers can exploit these backdoors, like, the Summer Baby Zoom WiFi camera comes with hardcoded admin access. The security researchers were able to exploit this vulnerability to compromise the device [77].

*2) **MAC/Adaptation/Network Layer**:* Numerous threats affect security at MAC layer, such as unfairness, interrogation, impersonation and Sybil attacks [78, 79]. Some of the DoS attacks at this level include collision

TABLE II – *Threats to IoT*

Ser	Threat	Vulnerabilities Exploited	References
Generalized Threats			
1.	Eavesdropping and traffic analysis	Lack of encryption and network access control	[40]
2.	Masquerading and unauthorized disclosure of personal information	Weak data security, authentication and authorization mechanism	[41]
3.	Device integrity	Lack of physical security, no tamper-proofing, trustless environment,open physical interfaces, boot process vulnerabilities	[42, 63]
4.	Remote code execution	Lack of host-based or strong network level security	[43]
5.	Software/Code integrity	No malware detection mechanism, weak network and application layer security	[9, 8]
6.	Threats to communication protocols (MITM,unauthorized access, DoS	Spoofing the ARP, brute-forcing pre-shared WiFi key, vulnerability in the exchange of disassociation message	[45, 47, 53]
7.	DoS (Resource exhaustion) attacks	Weak network and application layer security	[23]
Physical/Perception Layer			
1.	Eavesdropping	Unprotected communication channel, no encryption	
2.	Loss of power / Battery drainage	Unchecked volume of legal requests, lack of spam control	
3.	Hardware failure/exploitation	Negligence by the manufacturers, Faults of developers (hardware and software), Unprotected interfaces (e.g., UART, JTAG), weak application/web,network security	[8, 67]
4.	Malicious data injection	Weak access control	[68]
5.	Sybil attack	Lack of identity and device management	[69]
6.	Disclosure of critical information	Lack of physical protection for the devices	
7.	Side channel attacks		
8.	Device compromise	Vulnerable physical interfaces, Boot process vulnerabilities	[71, 72]
9.	Timing attacks and hardware exploitation	Open debugging ports	[1, 73]
10.	Node cloning	Lack of hardware security standardization and tamper-proofing	[74]
11.	Semi-invasive and invasive intrusions	Lack of physical security and tamper-proofing	[75]
12.	Change of configuration/Firmware-version	Weak implementation of cryptographic algorithms	[75]
13.	Unauthorized access to the devices	Use of default or hardcoded username and passwords	[67, 77]
MAC/Adaptation/Network Layer			
1.	Unfairness, impersonation and interrogation attacks	Weaknesses in communication protocols (channel access scheme), MAC spoofing, weak network access control,	[78, 79]
2.	DoS attacks to include collision attack, channel congestion attack, battery exhaustion attack, exploitation of CSMA, PANId conflicts	Flaws in medium-access control and communication protocol	[10, 30, 41, 79, 80, 81]
3.	Fragmentation attack	Lack of security mechanism in 6LoWPAN	[24, 82]
4.	MITM, eavesdropping	Weak authentication and data security	[68]
5.	Spoofing, hello flood and homing attacks	Weak authentication and anti-replay protection	[10, 83]
6.	Message fabrication/modification/replay attacks	Weak data authentication and anit-replay protection	[68, 84]
7.	Network intrusion and device compromise (remotely using malware)	Weak network intrusion detection/prevention system, weak device access control once the device is operational,inefficient identity management	[8, 85]

Continued on next page

TABLE II – *Continued from previous page*

Ser	Threat	Vulnerabilities Exploited	References
8.	Node replication attack and insertion of rogue devices	Weak network and device access control	[78, 86]
9.	Selective forwarding attack, Sybil attack, wormhole attack, blackhole attack	Weaknesses in network routing protocols	[10, 87]
10.	Storage attacks	Centralized data storage, non-replication of data storage, no protection against malware such as cryptlocker and ransomware	[8]
11.	DoS attacks launched by sending fake/false messages to a node, server or a gateway device	Weak link layer authentication and lack of anti-replay protection	[41, 83, 88]
Application Layer			
1.	Malicious codes	Lack of application/web security, authentication and authorization mechanism	[8]
2.	Software modification	Lack of application/web security	[9]
3.	Brute force and dictionary attacks, escalation of privileges and data tampering	Weak authentication and authorization mechanism	[89]
4.	SQL injection attacks	Injection flaws in SQL/noSQL Databases, OS and LDAP (Lightweight Directory Access Protocol)	[90]
5.	Identity theft and password/key/session-token compromise	Incorrect implementation of authentication in applications vis-a-vis session management	[89]
6.	Disclosure of sensitive/private data	Insecure web applications and APIs	[89]
7.	XSS (Cross-Site Scripting)	Vulnerabilities in web applications and user unawareness	[91]
Semantics Layer			
Identity theft, compromise of user privacy		Lack of data/application security	[92]

attack, channel congestion attack [10, 80], battery exhaustion attack (by increasing the frame counter value and spoofing of acknowledgement frames) [30, 81], exploitation of Carrier Sense Multiple Access (CSMA) by transmitting on multiple channels [30, 80] and initiation of fake PANId conflicts. At Adaptation layer, there is a likelihood of a fragmentation attack on 6LoWPAN protocol [24, 82].

Next, comes the Network Layer, at which most of the attacks are anticipated because it not only connects multiple private LANs to each other but also provides an interface to the internet. Significant threats to security and integrity of the system include MITM, eavesdropping [68], spoofing [10], message fabrication/modification/replay attacks [68], unauthorized access to network, compromise of a device (done remotely using malware) [8], node replication [78] and insertion of rogue devices [86]. Similarly, the threats to the availability of the network/services are; hello flood attack, selective forwarding, Sybil attack, wormhole attack, blackhole attack [10] and storage attacks [8]. DoS Attacks can also be launched by sending fake/false messages to a node, server [41] or a gateway device [88].

3) ***Application Layer****:* Security is never a preference for the application developers rather they focus more on efficiency and service delivery. As a result, the applications can easily be compromised, and their services can be denied to the legitimate users. Major threats to application layer are:

Malicious Code. Malicious codes spreading over the internet or targeted malware can easily compromise the connected IoT devices by exploiting their unique vulnerabilities, e.g., lack of application security and weaknesses in authentication and authorization mechanism. The infected devices can be used as bots to launch further attacks on other end devices/network applications [8].

Software Modification. An attacker can compromise an IoT device physically or by remote access and then modify the software or firmware to perform an unauthorized action [9]. The exploitation can be done via binary patching, code substitution or code extension.

Weak Application Security. Security of application/OS running on an IoT device is of utmost importance. Any weakness in the authentication and authorization mechanism can result in brute force attack, dictionary attack, unwanted disclosure of information, elevation of privileges

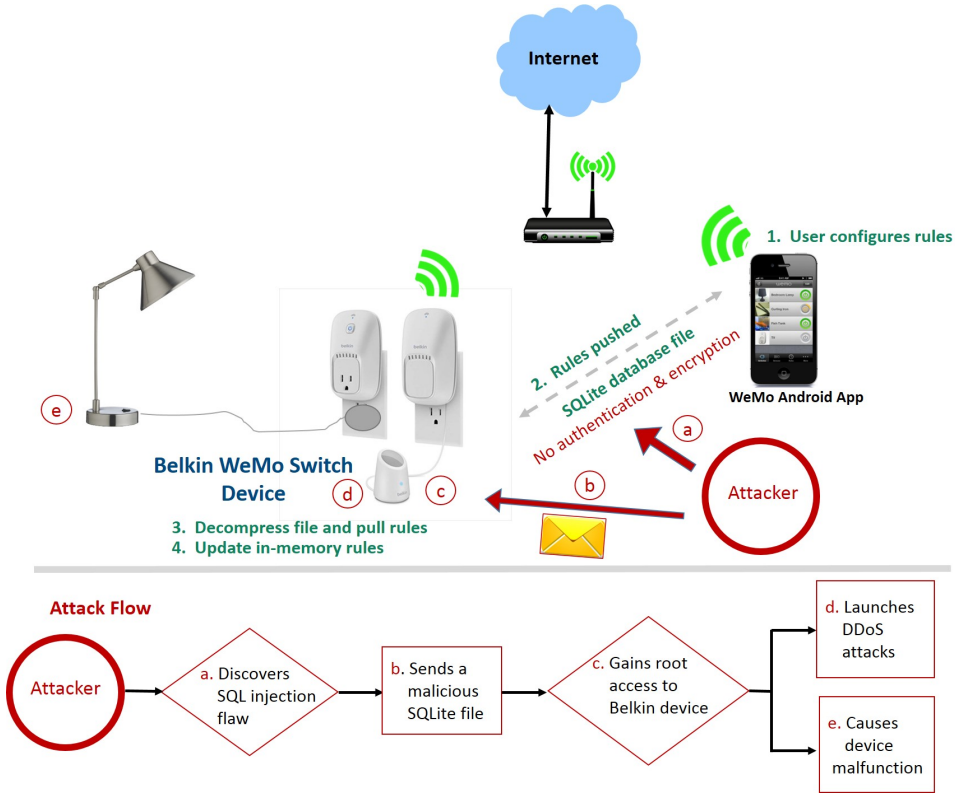

Fig. 6: Attacking a Belkin WeMo Switch by Exploiting an SQL Injection Vulnerability

and data tampering. Moreover, the latest application security risks ranked by Open Web Application Security Project (OWASP) [89], pose a valid threat to IoT systems that rely on websites and applications to provide relevant services to their users. Some of these application risks include:

- Injection flaws that threaten SQL/noSQL Databases, OS and LDAP (Lightweight Directory Access Protocol), pose an equal risk to IoT application and database servers. In such an endeavor security researchers were able to exploit an SQL injection vulnerability in Belkin's smart home products [90]. This vulnerability allows an attacker to inject malicious code into the paired Android WeMo smartphone app, and take root control of the connected home automation device. As shown in Figure-6, the sequence of attack is illustrated in 5 steps, i.e., from a to e. In that, firstly, the attacker discovers an SQL injection vulnerability in the Belkin WeMo Android app. He also discovers that there is no authentication and encryption used for communication with the Belkin device. Hence, anyone can send a malicious SQLite file to the device. He does the same and resultantly gets root level access to the Belkin device. Once inside, the attacker can launch a DDoS attack or can cause the IoT devices to malfunction. E.g., The lamp is kept on for a long time irrespective of the rules defined by the user. It is imperative to mention here that once an attacker gains root level access to the device; he can even kill the firmware update process initiated remotely by the vendor. Hence, the device can be kept in the compromised state for as long as desired by the attacker or until the device is updated on site.
- Incorrect implementation of authentication in applications vis-a-vis session management allow attackers to steal IDs of other users and compromise passwords, keys, and session tokens. The inability of a user to change the default username and password to access a new device or application is an example of this weakness. This aspect is critical for IoT systems based on smart devices, such as smart city, smart home, smart vehicles and wearable health monitors. An example of such a vulnerable device is The Withings Smart Baby Monitor that allows the users to monitor their babies remotely via a mobile app. However, the video stream sent from the baby monitor to the WiFi Router is in plain-text. Hence

researchers in [25] were able to hijack the session using ARP poisoning and gain access to the camera feed.

- Sensitive data exposure due to insecure web applications and APIs pose a threat to the confidentiality and privacy of user data collected or processed by IoT devices such as smartphones, wearable health monitors and smart watches. An example of such a vulnerability is the Philips Hue Smart Bulb [25]. It enables the user to control the lighting system through a mobile app wirelessly. However, the data exchange via HTTP between the app and the ethernet-enabled bridge that forwards the commands to the smart bulb is in plain text. Hence, any MITM attacker or eavesdropper can sniff the communication between the user and the smart bulb and ascertain personal habits of the user. Moreover, attacker can also extract the list of authorized users from the bridge and can masquerade as a legitimate user later. The threat scenario is shown in Figure-7.
- Broken access control is due to lack of restrictions on authenticated users. Same can be exploited in an IoT system by attackers to access unauthorized functionality or data. Such as change of health monitor's thresholds for generating an alarm/notification.
- Security misconfiguration is one of the most common weaknesses. It implies insecure default configurations, open cloud storage, misconfigured HTTP headers, and overblown error messages that may contain sensitive information. An IoT device is insecure without secure configuration and timely upgradation of its OS and applications.
- XSS (Cross Site Scripting) is a very prominent threat to web-based applications, and IoT is not an exception. Correspondingly, security researchers were able to exploit a XSS vulnerability in Belkin's smart home products [90]. Such a vulnerability allows an attacker to run an arbitrary JavaScript code in the victim's browser [91]. It can further lead to hacking into the phone and theft of private data.

4) ***Semantics Layer****:* The creation of semantics web has transformed the web from human-readable form to machine processable form. The machine processing has no doubt augmented the human reasoning, interpreting and decision-making abilities based on automated Big Data analytics. However, extraction of intelligence or application specific information from Big Data has its security and privacy issues. E.g., unauthorized disclosure of personal information stored on social media or sensitive health-related data may compromise privacy of a user. Currently, the tools being used to store and compute big data, such as HDFS (Hadoop Distributed File System) and Google's MapReduce framework are considered inadequate to protect sensitive data [92].

E. ***Security and Privacy Challenges to Cloud-Supported IoT***

The vision of future IoT is a large-scale integration of various technologies, i.e., sensors, actuators, personal devices such as smartphones, location services, applications, servers, etc. The data originating from a multitude of devices will be available for open sharing across a range of applications, servers, and users. This public sharing is currently achieved with the cloud technologies. Over the period cloud computing [93] has evolved to process, analyze and store Big Data. Though, cloud services offer benefits in terms of resource management, scalability [11, 94], cost effectiveness and shifting of business risks including hardware failures to the infrastructure providers that have better risk management capabilities [95]. However, mostly the IoT systems are developed for a particular application in mind. Therefore, the security aspects are also limited to that particular application with very less or no consideration for security while data is in the cloud and being shared openly across a range of devices. If the legacy IoT systems are connected with the cloud for extended data sharing, i.e., horizontally between things or various applications via the cloud, the IoT subsystems usually consider and adopt security measures within their sub-networks. However, once the data leaves the sub-group and enter the cloud for wide/open sharing, then numerous issues of security and data privacy emerge. In addition to data confidentiality there are other issues in cloud computing concerning trust mechanism between the service provider and cloud infrastructure provider at various layers of cloud architecture [95].

1) ***Security of Data****:* The cloud usually provides secure communication using TLS/DTLS (Datagram Transport Layer Security). TLS provides communication secrecy (using symmetric key encryption), server authentication (using Public Key and Domain Controllers) and message integrity using MAC. Now here a question arises that what if the data is encrypted by the things before it is sent to the cloud? This encryption by things will have following impacts:

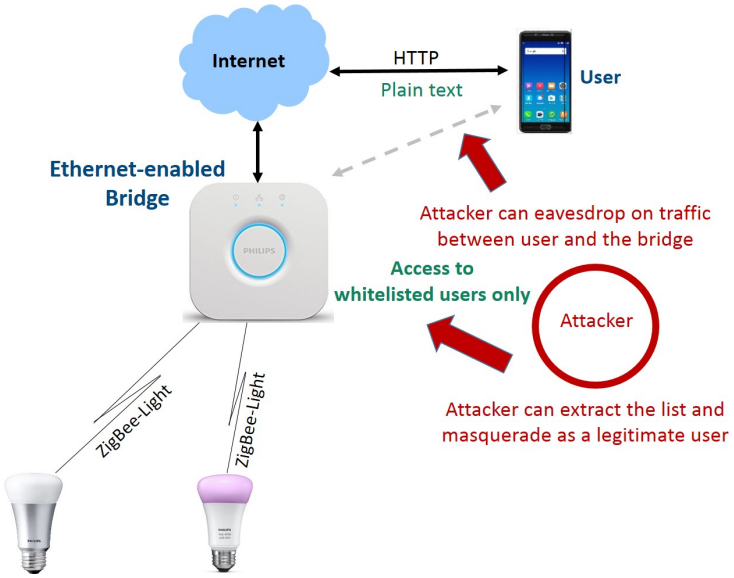

Fig. 7: Threats to Philips Hue Connected Bulb

- The Cloud provider will not have access to legible data.
- The data cannot be shared publicly.
- The security is to be managed by the things including complexities of key management, especially, once the old keys are revoked, and new keys have to be generated and issued.
- It will affect scalability and restrict data aggregation and analytics to be performed by the cloud provider.
- Cloud provider is restricted to provide only storage/IaaS (Infrastructure as a Service).

2) ***Handling of Heterogeneous Data****:* IoT applications deal with large amount of widely distributed data gathered from sub-systems based on multitude devices like WSN, RFID, smartphones, GPS, etc. Such a diversified data may exist in different formats hence demanding appropriate data fusion before the cloud can analyze it. However, integration and fusion of such a heterogeneous data may create privacy-related issues [68].

3) ***User Anonymity Vis-a-Vis ID Management****:* In a cloud-supported IoT, drawing a balance between user anonymity and ID management for authentication, authorization, and audit is a big challenge. E.g., in eHealth applications, the health-related data of patients is provided to various organizations for data analytics and development of future policies on health issues. Importance of such a use of patient data for improving health care cannot be denied. However, it always raises security and privacy concerns for the patients. Hence, various user anonymity techniques are being practised to disassociate the ID of the patients from the health data. But at the same time, to ensure the security of the cloud-based health services, user authentication is equally essential for restricting network access to the legitimate users only.

4) ***In-Cloud Data Sharing****:* The vision of future IoT is extensive sharing of data across a range of devices and applications, which can only be achieved with a policy on protection and sharing. Otherwise, if things' data is stored on the cloud and isolated from other devices [94], the data processing incorporating multiple streams may not be possible, and it may also affect the efficient data analytic services by the cloud provider. Furthermore, it is estimated that at least one-fifth of the documents uploaded to file-sharing services contains sensitive information and 82% of cloud service providers ensure data security during transmission. However, only 10% encrypt data, once it is stored in the cloud [17].

5) ***Large-Scale Log Management****:* In a cloud-supported IoT, there would be a huge number of heterogeneous devices such as sensors, smartphones, smart controllers, etc. Therefore, logging and audit of the network may be challenging. Few of these challenges may include: What does the cloud provider must record? If the log is decentralized then there would be variations in what is recorded on different systems, and resultantly there would be different interpretations of the logged data [96]. Moreover, insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, may result in implausible auditing and accountability thus allowing attackers to launch further

attacks on the systems. Hence, no doubt most breach studies show that time to detect a breach is over 200 days which is typically detected by external parties rather than internal processes or monitoring [89].

6) ***Vulnerability To DoS Attacks****:* Cloud providers usually implement requisite controls to protect against various cyber-attacks. These checks include vulnerability mitigation by updating the OS, secure computing using TPM to protect against malware/code modification attacks, etc. Even if an attack is successful, the isolation mechanisms contain the effects. However, an IoT Cloud is vulnerable to a DoS attack launched from compromised things. Moreover, cloud services are usually designed to scale up/down resources in response to increasing demand but are still vulnerable to DoS attacks [97].

7) ***The Threat of Malicious Things****:* The cloud being resourceful and the coordinator between things can augment the security of cloud-based IoT systems. It can detect a malicious thing/node during the validation process. The cloud can also offer a protective security measure by triggering software/firmware updates where deemed necessary and resultantly sending control messages to the things to revoke them from the network or turn them off. However, there are some challenges involved in determining/detecting the malicious nodes in a system [98]. These problems may include: What method be used to identify or detect a malicious node? When to initiate the node attestation procedure? If the attestation is based on software/code verification, then will it be a challenge-response protocol or a one-way attestation scheme? Finally, is software-based attestation scheme effective? or there is a need for a hardware-based attestation protocol.

F. ***Security and Privacy Issues in Fog Computing for IoT***

Cloud security is an important factor that has adversely affected the development of cloud computing. Cloud's centralized data storage and computing framework present a single point of failure and a concentrated target to the attackers. Hence, to reduce the visibility of end nodes to the external attackers, fog computing enables the data to be transiently maintained and analyzed on local fog nodes thereby, also reducing the processing load, overcoming the bandwidth constraints and minimizing the latency for time sensitive applications in IoT [100, 101]. Fog computing does compliment the cloud by reducing the latency in data provisioning [102], however, as it is deployed by different fog service providers that may not be entirely trusted the devices are vulnerable to be compromised. Fog nodes have distinctive features, such as decentralized infrastructure, mobility support, location awareness and low latency [103], which make them vulnerable to various security and privacy threats [104, 105]. These threats include identity and data forgery, eavesdropping, MITM attacks, DoS attacks, data and device tampering, Sybil attack and user privacy leakage (identity and location information, social habits, personal details etc.).

Although all the threats discussed in preceding sections require due attention. However, the primary objective of this paper is to get the attention of security researchers to one of the most realistic and currently practised issue of code modification and malware attacks. Which, if left unattended will prove detrimental to the security of future autonomous IoT systems.

Correspondingly, Bruce Schneier, Chief Technology Officer (CTO) at IBM Resilient states that IoT devices being connected to the internet are vulnerable to ransomware attacks [106]. Recently, in a practical demonstration of such an attack, white hat hackers have developed a first of its kind ransomware that compromises a smart thermostat and then demands a ransom to unlock it [107]. Such a demonstration has shown the possibility of remote code execution on smart devices that can ultimately compromise the complete network, e.g., smart home, smart grid, ICS, smart city. It is, therefore, imperative to understand the malware attack and its methodology, to prepare a strong defense.

III. MALWARE THREAT

The history of computer viruses goes back to 1981 when the first “In the Wild” computer virus named Elk Cloner targeted Apple-II systems [108]. Moreover, since the commercialization of the internet in the early nineties, there has been a considerable rise in cyber-attacks around the world. This number has drastically increased since the start of the twenty-first century. Same can be observed in Table-III that shows the trend in different types of malware over past thirty-seven years [109, 110]. IoT devices being connected to the internet are equally vulnerable to malware attacks. Hence, it is essential to analyze the functioning and attack methodology of some of the significant malware.

TABLE III:
Trending in Malware Attacks

Malware Type	1981-1990	1991-2000	2001-2010	2011-2016	2017	2018
Virus	10	07	03	-	-	-
Worm	01	02	27	01	-	-
RAT + Rootkit	-	-	21	12	-	1
Botnets	-	-	2	2	-	-
Ransomware	1	-	-	16 [99]	2	-
Total	12	9	53	17	2	-

A. ***Anatomy of Malware***

Different types of malware are developed to achieve diverse objectives. Some are research-oriented, and some are released into the wild to attain malicious aims set by the attackers. The malware roaming in the wild can further be categorized as targeted and general threats. Before we go further, it is imperative to clear the difference between a threat and an attack. In information security domain, a threat can be defined as a constant danger that has the potential to cause harm to an information system, such as malware, application misconfiguration, and humans. Whereas, an attack is the successful execution of a malicious act by exploiting vulnerabilities in an information system. Therefore, in this section, an attack methodology of some of the successful malware attacks is explained. Although a plentiful of malware attacks such as NotPetya [111], DuQu-2 [112, 113, 114, 115], Cryptlocker [116], Shamoon-1 [117, 118], Shamoon-2 [119, 120], Flame/SKyWIper [121, 122, 123, 124], Gauss [123, 125, 126], Icefog [127], Dragonfly-Group/Energetic Bear [128, 129], Red October [130, 131, 132], and Night Dragon [121, 133] have been analyzed to derive the attack methodology (discussed in Section-III.B). A perceived attack sequence of a cyber-attack based on a malware is shown in Figure-8. However, detail of some of the significant malware attacks targeting IoT systems including ICS, CPS, smart devices, and critical infrastructure is mentioned here. The attack methodology amply covers the attack description, vulnerabilities exploited, attack vectors, propagation mechanism, and effects incurred by respective malware.

1) ***Xafecopy Trojan****:* A Trojan from Ubsod family (Blue Screen of Death) was identified in Sep 2017 by Kaspersky Labs as Trojan-Clicker-AndroidOS.Xafekopy [134]. Xafecopy trojan mostly disguised as a battery optimizer app targeted WAP (Wireless Application Protocol) based Android devices. The malicious app subscribes the victim user's MSISDN (Mobile Station International Subscriber Directory Number) for numerous services on various websites with WAP billing system that charges directly to the user's mobile bill. This trojan is also capable of bypassing the CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) systems. A modified version of Xafecopy can also send SMS from the user's phone to some premium-rate phone numbers. They can also delete incoming SMS from the mobile network provider, and hide notifications about balance deduction by checking for words like "subscription" in the incoming messages. It is also capable of switching a user from WiFi connection to mobile data.

2) ***WannaCry****:* A typical ransomware also known as, Wanna Decryptor, WannaCrypt, WanaCrypt0r and WCry [135] was detected in May 2017. By then it had affected 230000 systems including health, telecommunications, transportation, shipping and energy sectors in 150 countries. It propagated over the internet and exploited Server Message Block (SMB) vulnerability (MS17-010) in Microsoft Windows 7, 8, 10 and XP systems. It is assumed that it probably spread through phishing emails or malicious websites [136]. Once inside the target system, it would encrypt selected file types before deleting the original files. The malware also changed the windows wallpaper and displayed a message bearing instructions on how to make the payment in Bitcoins to get the files decrypted. The worm had a killer switch in itself as it looked for a non-existent domain (www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com) to continue exploitation. However, a security expert found out this weakness and created the respective domain thus slowing down the propagation of the malware [137].

Moreover, security researchers in [138], have identified that the ICS is of primary concern in the backdrop of malware, especially ransomware attacks. It is because most of the ICS are always in an operational state, hence, it is difficult to patch them. Moreover, the ICS software and protocols rely on NetBIOS and SMB (Server

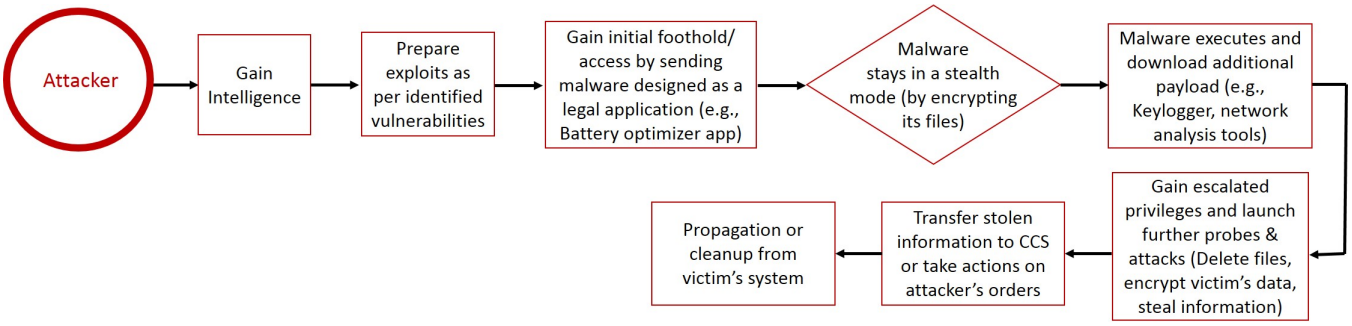

Fig. 8: Malware Attack

Message Block) for operation in a distributed computing environment. Therefore, malware exploiting SMB and NetBIOS vulnerabilities can cause an adverse effect on these systems.

3) ***Cryptlocker****:* Researchers discovered four million samples of this ransomware in 2015. Cryptlocker encrypted files on the target systesm thus restricting access of legitimate users to their data. The objective was to get ransom in return for decrypting the data [116]. The attackers used Angler Exploit Kit to find the vulnerabilities that were exploited by the malware. The malicious software is embedded in a pdf document and propagates as an email attachment through Gameover Zeus Botnet using encrypted peer-to-peer communication named Kademlia [139]. It is installed in the user profile folder %APPDATA% or %TEMP%. The vulnerable systems and applications include windows, MAC, Linux, internet explorer and Adobe Flash. Cryptlocker kept its files encrypted which made it difficult for ordinary users to identify the malicious files. Moreover, to avoid forensics, the malware clears itself from the target computer after putting up ransom demand. It is estimated that Cryptlocker inferred a loss of over USD (US Dollars) 1 Billion in 2016. The gravity of such an attack can be ascertained from an incident in Austria [140], where an electronic lock system installed in a hotel was attacked, and guests were locked out of their rooms. The hotel management had to pay 1500 Euros as a ransom to get the system unlocked by the attackers.

4) ***Mirai****:* An internet based DDoS attack [44] launched against a computer security journalist Brian Krebs's security website through IoT Botnets created out of DVR (Digital Video recorders) and CCTV cameras. The IoT Botnets directed 620 Gbps traffic towards the website. The attackers exploited the default username and passwords hardwired on the DVRs and CCTV Cameras to gain access to these devices by launching a dictionary attack involving sixty-two default usernames and passwords for various account types, such as root, admin, guest, and service. Same malware was also involved in an attack on a French Cloud Computing Company "OVH" [141] and an attack on a DNS provider Dyn in October 2016. The attack on Dyn affected services of some of the significant technology, eCommerce and web giants in the world such as Amazon, Airbnb, PayPal, Visa, Twitter, HBO, CNN, and BBC.

5) ***Havex****:* Also known as "Backdoor: W32" and "Havex.A", is an ICS focused Remote Access Trojan (RAT), created with an objective of spying on the infected host/server. It targeted websites of three ICS vendors. It also has the potential to cause a DoS Attack on OPC (Open Platform Communications) based applications [142]. Attackers used three attack vectors to entice the victims to install the software on their systems including spam emails, exploit kits and use of watering hole attacks, i.e., software installers on prominent vendors' sites were infected with RAT thus any user downloading the software or an update would automatically download and install the Trojan. The malware exploited the vulnerabilities in vendors' websites to Trojanize the software installer. The Trojanized installer comprised a malicious file named "mbcheck.dll", which was the actual malware. This file was dropped and executed as the part of a standard installation. RAT would then communicate with a Command and Control Server (CCS) and download numerous plugins for further attacks. Various versions of RAT plugins had different tasks like enumerating LAN and listing down connected resources and servers using OPC [143].

6) ***Stuxnet****:* A targeted computer worm designed to sabotage CPS (Cyber Physical System) installed in Iranian Nuclear Enrichment Facility was discovered in

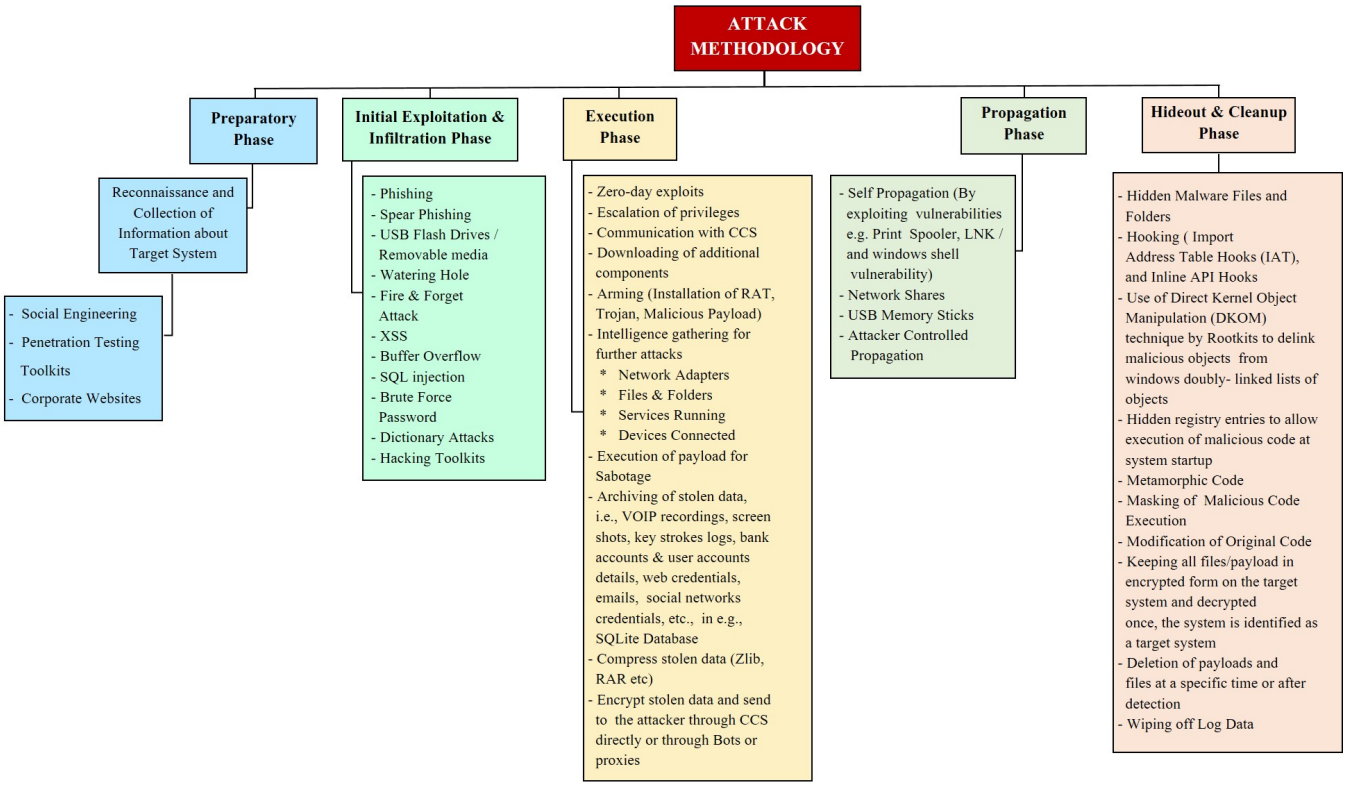

Fig. 9: Methodology of a Malware Attack Targeting IoT/ICS

2010. It was delivered through an infected USB flash drive. Stuxnet exploited four zero-day vulnerabilities in Windows-based systems to gain an initial foothold. It's payload was designed primarily for ICS. The malware consisted of multiple modules including Windows and PLC rootkits, anti-virus evasion techniques, complexed process injection and hooking code, network infection routines, peer-to-peer updates and a CCS interface [144]. Stuxnet specifically targeted PCs running WinCC/PCS-7 control software used for programming the PLCs [145]. It could act as a MITM attacker and mask the malicious code execution by replaying 21 seconds of legitimate process input signals. It had the capability of self-propagation by exploiting print spooler and LNK vulnerability (CVE-2015-0096) in Windows. It comprised rootkits which could hide its presence and was also equipped with stolen digital certificates to appear legitimate. The payload altered the frequency converter drives (from specific vendors Fararo Paya from Iran and Vacon from Finland) speed to cause physical damage to over 900 centrifuges [146]. Microsoft released a security update MS10-061 to fix print spooler and MS-15-018 for windows shell vulnerability.

B. ***Attack Methodology***

It can be deduced from different characteristics of malware discussed in the previous section that in the last decade or so, malware attacks have not only affected the IT infrastructure but have caused physical damage to ICS as well. Security researchers in [138], have identified that the ICS are of primary concern in the backdrop of malware, especially ransomware threats. It is because most of the ICS are always in an operational state, hence, it is difficult to patch them. Moreover, the ICS software and protocols rely on NetBIOS and SMB (Server Message Block) for operation in a distributed computing environment. Therefore, malware exploiting SMB and NetBIOS vulnerabilities can cause an adverse effect on these systems. Hence, keeping in view the operating mechanism and functionalities of the malware, we have formulated an attack methodology (shown in Figure-9). It illustrates all possible steps taken by the attackers in various phases to attack an isolated/public IoT network remotely.

1) ***Preparatory Phase****:* In this phase, attackers carry out reconnaissance and collect information about the potential target. The information can be obtained through social engineering, corporate websites and by using

various penetration testing toolkits such as Metasploit, Wireshark, Nmap, Social Engineering Toolkit, Kali Linux, and Nessus. The penetration testing is done to find the weaknesses in the target system. The testing can be performed on networks, websites, and servers. Based on this information, attackers plan their attack vectors and develop the malware.

2) ***Initial Exploitation and Infiltration Phase****:* After gaining information about the potential target, the attackers decide on the type of exploit, its functionalities, and the attack vectors to deliver the exploits to the target systems. In most of the organizations, not only administrative staff but even the technical staff is not sound on information security issues. Therefore, attackers utilize this weakness and resort to phishing, spear phishing, watering hole attack and use of infected USB flash drives to gain an initial foothold in the target systems. There are some other exploitation methods as well, such as Cross Site Scripting (XSS), buffer overflow, SQL injection, brute force and dictionary attacks for password recovery and use of hacking toolkits.

3) ***Execution Phase****:* After intruding into the target system, the attackers can steal information or perform a malicious action either by remote access or through a sophisticated malware that installs a Trojan on the host system. The malware can be installed by exploiting zero-day vulnerabilities for which no security update is available, or by attaining root/admin privileges. Most of the latest malware versions keep their files in an encrypted format to avoid detection by anti-virus or any other security mechanism. As soon as, the malware identifies the target system based on the particular file system, filename keywords, pathname or some other attributes, the payload is decrypted and executed.

In many cases, the payload installs a RAT, which then communicates with a CCS and downloads additional components of the payload or other toolkits/exploits. Some of the functions a RAT performs include intelligence gathering on network adapters, files and folders, services in operation, and connected devices. In addition to espionage, a RAT can enable an attacker to perform any function on the host system from the escalation of privileges to physical damage to the hardware. The RAT is also capable of archiving the stolen data files, VOIP recordings, key logs and financial information. The current breed of RATs uses SQL Lite Database, that archives the data in a compressed format. The stolen data is usually encrypted before being sent to the CCS. The data may be delivered directly to the CCS or through bots to increase complexities for later forensics. Some of the most notorious RATs currently in use are; Sakula, Sub7, KjW0rm, Havex (specifically for ICS), ComRAT (Targets ICS networks), Heseber BOT, Dark Comet, and Shark.

4) ***Propagation Phase****:* The common attribute in both, ”Targeted” and ”In the Wild” malware is the capability to reproduce or to move from the infected system to a new host. Because of this functionality, the malicious software is also termed as self-propagating malware. These malicious programs exploit security vulnerabilities at various levels, i.e., application layer, network layer and web servers to infect systems and then scan the internet/LAN for more vulnerable systems. Such weaknesses include print spooler, LNK/Windows-shell vulnerability, network shares and USB memory sticks. The installation of RAT also facilitates attacker controlled propagation in the victim network.

5) ***Hideout and Clean-up Phase****:* Malware use multiple techniques to keep themselves invisible, while operating on a victim system. Usually, they keep their files and folders hidden or keep them encrypted. The encrypted files are decrypted once the malware reaches the target system or at the time of execution. Malware, such as rootkits remain invisible by faking the output of API calls through hooking techniques. The hooking can be achieved by intercepting function calls, altering import tables of executables and use of a wrapper library. Two most common methods of hooking being implemented by malware are Import Address Table (IAT) Hooks and Inline API Hooks. The rootkits also resort to Direct Kernel Object Manipulation (DKOM) technique that hides its processes, drivers, files, and intermediate connections from object manager/task manager. For clandestine operation, these sophisticated malware are also capable of making hidden registry entries to allow execution of malicious code at system startup. To remain undetected from anti-virus, the malicious software are designed to be metamorphic, i.e., to re-write their code after each execution. In addition, to avoid forensics and reverse engineering, these malware can delete their payload and files at a given time or attacker controlled instances. They are also capable of removing log data to wipe-off their footprints.

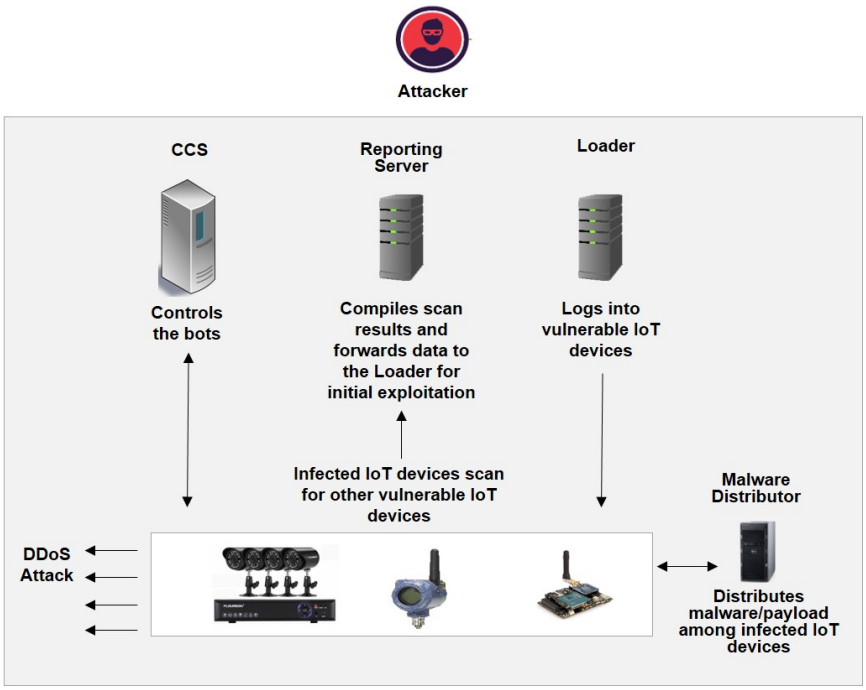

Fig. 10: IoT Botnet

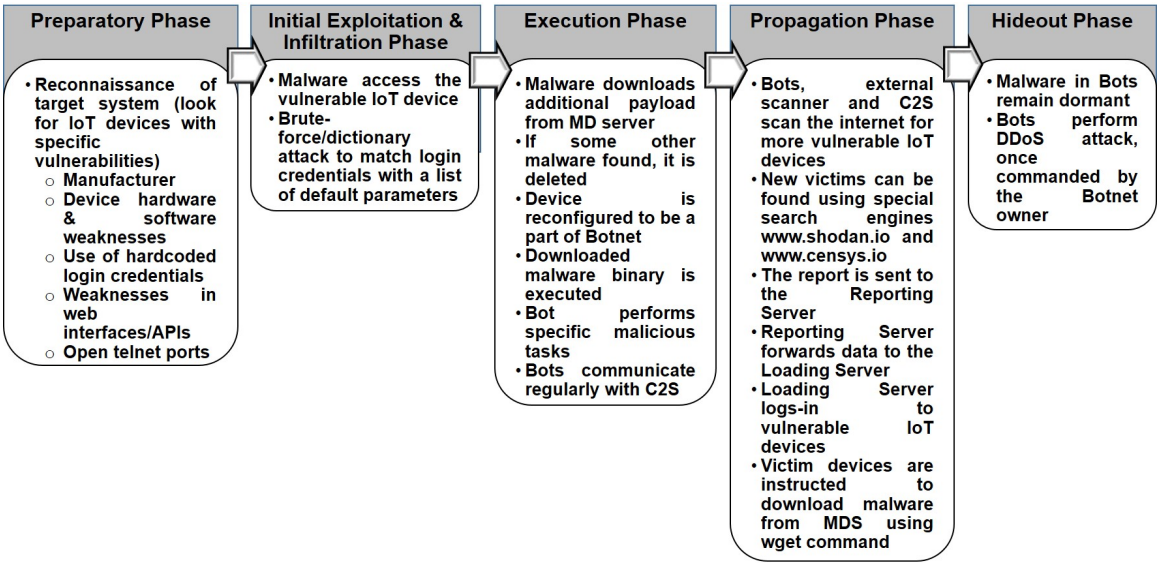

Fig. 11: DDoS Attack on IoT

IV. Gap Analysis and Security Framework

An exponential increase in the number of IoT devices is expected in next four years. However, due to lack of secure architecture and weak security mechanism in commercial IoT devices, these will continue to be a lucrative target for the attackers. Keeping in view the latest trends in malware-based cyber-attacks, there is a high probability that IoT devices may be used to create a botnet army to launch various other attacks such as DDoS and distribution of ransomware/spyware. Based on malware attack methodology described in Section-II, we have deduced an attack methodology of a DDoS attack on IoT devices, which turns the victim devices into bots. One of the probable architecture of a botnet controlled by an attacker is shown in Figure-10. A typical IoT botnet [147] comprises a CCS that controls the bots, a Reporting Server that compiles the data about vulnerable IoT devices and forwards it to the Loader module. The Loader gains an initial foothold into the

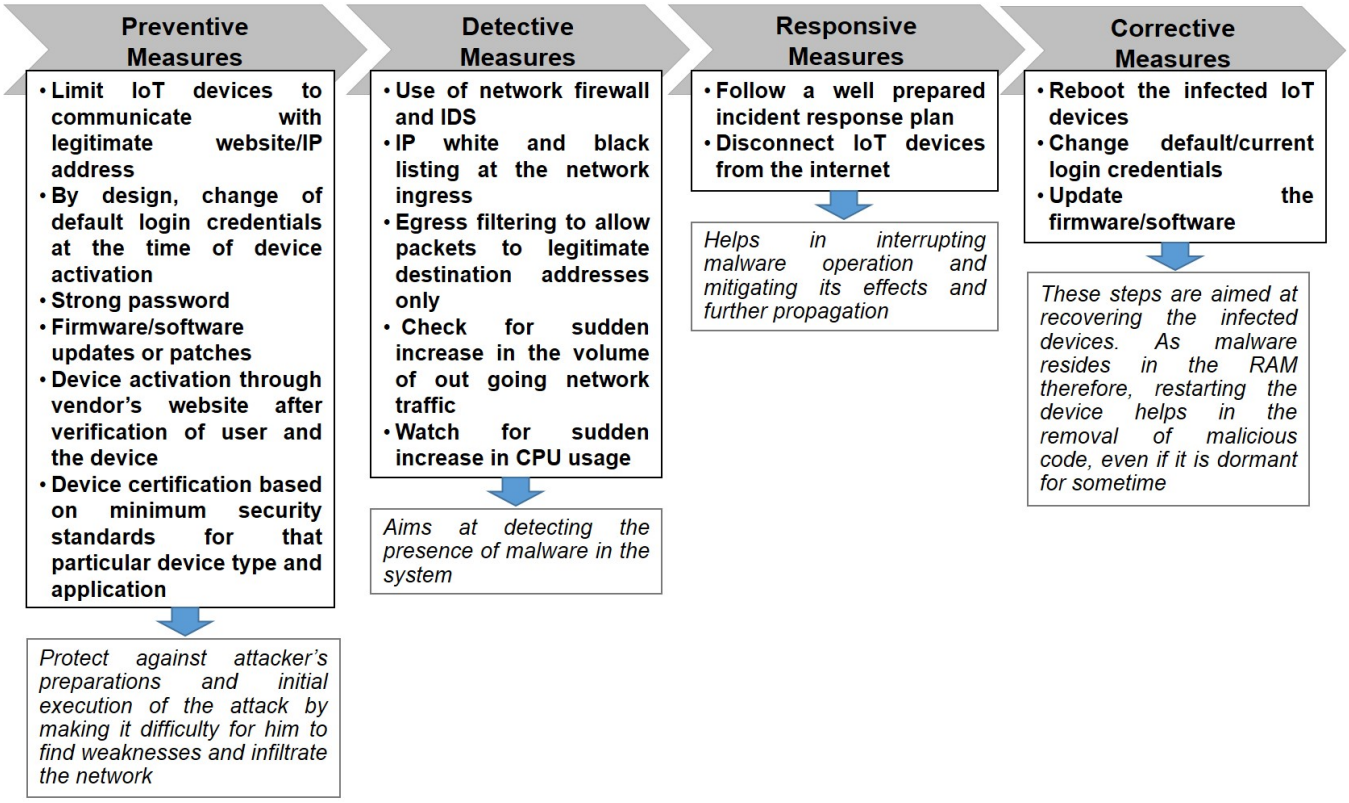

Fig. 12: IoT Security Against DDoS Attack

victim devices by exploiting the weaknesses such as hardcoded default login credentials. Once the Loader logs into the victim device, it instructs the victim device to contact the Malware Distributor (a server in the botnet) to download additional malware payload. The infected IoT devices such as CCTV cameras, DVR, smart meters or sensing nodes are then used to launch DDoS attacks. The chronology of this DDoS attack is shown in Figure-11.

In the preparatory phase, the attacker carries out the reconnaissance and find out specific vulnerabilities in IoT devices. The vulnerabilities may include, open hardware ports (UART, JTAG etc.,), weaknesses in the software/OS of the device, weak security implementation, i.e., hard-coded login credentials, weaknesses in the web interface or APIs, and last but not the least open telnet ports. After gaining information about IoT device's vulnerabilities, the attacker plans to get an initial foothold into the vulnerable devices by selecting/preparing appropriate exploits. In this case, the exploit can be in the form of malware, that establishes a telnet connection with the victim device and logs into the device by using brute-force or dictionary attack to find out the requisite username and password out of the list of probable default credentials that could be used by that specific device manufacturer.

In the execution phase, the infected IoT device downloads additional malware payload from the Malware Distributor. The malware scans the infected IoT device for other malicious codes, if found, they are deleted, and victim device is reconfigured to be a part of the IoT botnet. The IoT bot is then used to launch specific attacks such as the DDoS attack on targeted websites or servers. During their lifetime, IoT bots communicate regularly with the CCS and receive instructions for further attacks. The infected IoT devices also scan the internet or the internal network for vulnerable devices and send the scan results to the Reporting Server. In case of the internet, list of vulnerable devices can be found using specialized search engines such as www.shodan.io and www.censys.io. The Reporting Server forwards the list of vulnerable devices to the Loader module, which logs into the vulnerable IoT devices and then instructs them to download additional malware/payload. Usually, the additional payload is downloaded using wget command. The malware can remain dormant to hide its presence and performs the DDoS attack only when commanded by the attacker through

CCS.

Based on the above mentioned DDoS attack, which is just one of the numerous threats /attacks facing IoT, it is evident that current IoT security standards and protocols being implemented by the IoT device manufacturers fail to protect against modern era's sophisticated malware attacks. Although existing IoT communication protocols including CoAP, RPL, 6LoWPAN and 802.15.4 do provide communication security at various layers of the IoT protocol stack (shown in Table-IV). However, the communication protocols alone, cannot protect against malware/code-modification attacks [24, 30]. Hence, this paper proposes a security mechanism (shown in Figure-12) against IoT botnet malware, comprising preventive, detective, responsive and corrective measures. In addition to the security measures, the proposed security model also illustrates the impact on an attacker's methodology of attack based on various phases, as shown in Figure-10. However, in a realistic world keeping in view the plethora of IoT devices' vulnerabilities and related threats as discussed in Section-II, the proposed security mechanism as shown in Figure-12 is insufficient. Therefore, security of IoT ecosystem requires to be dynamic, innovative and wholesome so that it is always one step ahead of the adversaries. A comprehensive security mechanism means proactive approach towards the security of devices, data, applications, networks, and users. Hence, there is a need for concise and practical guidelines for the development of a dynamic IoT security framework that can detect contemporary threats, predict future security events and respond swiftly to mitigate the risks and take remedial actions.

A. ***Guidelines for IoT Security Framework***

To prepare a composite set of guidelines for edifying IoT security, we have reviewed the best practices currently being deployed by some of the technical giants of the world such as IBM (IBM Watson IoT), Cisco, AT&T (American Telephone & Telegraph), and TCG (Trusted Computing Group). A graphical illustration of these guidelines is shown in Figure-13 and Figure-14. Table-V also glances over the security measures and their respective impact/protection against threats. These security measures are discussed in details in the succeeding sections.

1) ***Risk Assessment and Threat Modelling****:* The first step in the development of a security policy for any organization is carrying out the risk assessment for all processes, equipment (hardware & software both), stakeholders and information assets at each layer of IoT architecture. E.g., starting from the manufacturing, transportation, installation and commissioning stage to the operation and management of the IoT system. The primary objective of this assessment is to identify what all security incidents can happen in the organization, and subsequently initiating the risk treatment process to minimize the damage of such events. Almost all the information security standards enforce risk management as an integral part of the overall controls.

ISO-27001 [150], an international standard for Information Security Management System (ISMS) outlines seven steps to an effective risk assessment. The first step is about How the organization is going to define its risk methodology? The methodology includes risk ownership, means of measuring the impact of risk on confidentiality, integrity, and availability of information and the method of calculating the effects of the identified risks. The second step involves determining all possible information assets, failure of which can cause some loss to the organization. The third step focuses on identification of threats and the potential vulnerabilities that can be exploited. In the fourth step, organizations are required to map risk impacts against the likelihood of their occurrences. The fifth step is the most important, as it involves the implementation of measures to avoid, mitigate, transfer or accept the risks. Sixth and seventh step includes preparation of risk treatment plan and continuous monitoring of the ISMS for any dynamic changes to the overall security plan. National Institute of Standards and Technology (NIST) have also issued a special publication 800-30 [151] as a guide to conduct a risk assessment for the security of information systems. Any such standard can be followed until there are some IoT specific standards on board.

2) ***Defense-in-Depth****:* Due to increase in sophistication and complexity of cyber-attacks, no IT infrastructure can be termed ”Safe”. No security measure claims to prevent 100% attacks. Therefore, the ”Defense-in-Depth” mechanism requires substantial preventive, detective, responsive and corrective actions. However, at the same time, implementation and practice of security measures should not be so complicated that users avoid and go around them. Hence, a comprehensive defense mechanism should be planned based upon risk profiles of the information assets of the organization. Cisco has issued a straightforward and handy defense in depth

TABLE IV:
Security Provided by IoT Communication Protocols

IoT Layer	Protocol	Security Measures
Physical	802.15.4	Nil [24]
MAC	802.15.4	Data Confidentiality, Data Authenticity & Integrity, Replay Protection, Access Control Mechanism [24]
Adaptation	6LoWPAN	Nil [24]
Network	RPL (Routing Protocol for Low Power & Lossy Networks)	Data Confidentiality, Data Authenticity & Integrity, Replay Protection, Semantic Security, Key Management [148]
Application	CoAP (Constrained Application Protocol)	Data Confidentiality, Data Authenticity & Integrity, Replay Protection, Non Repudiation [149]

strategy checklist [152] that can help in evaluating the overall security framework of an organization. Moreover, the defense in depth approach requires the organizations to take all possible preventive, detective, reactive and corrective measures. All of these steps are discussed in detail in subsequent sections.

3) ***Preventive Measures****:*

Security by Design. The architects of the IoT systems should consider the non-zero likelihood of device compromises while developing security protocols. Therefore, security should be enabled by design and users should have the leverage to change the security settings as per their requirements [17, 153]. In addition, due consideration should be given to the following points:

- The trusted environment for secure computing.
- Security of all open/debugging ports.
- The integrity of firmware/code.
- Access control based on multi-factor authentication.

Device Security. Allocation of a unique device identifier to each IoT device and its continuous validation is essential to ensure platform integrity and controlled access to system resources [154]. The devices should prove their unique ID to set up secure communication with their respective neighbors. The neighbor can be a node, a gateway device or an application server. The security of device ID against spoofing attacks is critical for sensitive organizations. Moreover, currently, device ID is required for most of the network security protocols such as IPSec, TLS, and SSH. Similarly, there should be some mechanism for safe storage of keys, passwords, certificates and other security critical information on the device, that cannot be tampered by the adversary [43].

To solve the problem of secure device ID, TCG proposes the use of TPM-based keys as device IDs, which complies with IEEE Standard for Local and Metropolitan Area Networks and Secure Device Identity (802.1AR) [155, 156]. The TPM provides enhanced security for device identifiers by protecting these keys in the hardware. Therefore, these keys are protected against unauthorized disclosure during malware and hardware tampering attacks. Another advantage of this technology is that being based on TPM, the cryptographic ID is bounded to the particular device [156], which makes it almost impossible for an attacker to spoof that particular ID using different hardware. However, it is a general opinion that use of cryptographic identifiers results in privacy issues. Therefore, to avoid long-term user keys/IDs that may lead to unwanted tracking, TCG proposes the use of TPM-based attestation identity keys or direct anonymous attestation.

There is also a requirement of device registration so that devices can be added or removed as and when required and only authorized devices are included in the network. The device registration may encompass maximum information about the device such as device ID, its role/capabilities, type, level of security/authorization as per sensitivity of data, public key, software/firmware version and authorized period of activation. One of the possibilities to ensure a transparent and immutable device registry is the use of Blockchain technology [43].

IoT devices often operate in an untrusted environment without any physical protection such as traffic light sensors, environmental sensors, agriculture sensors, smart city sensors and a lot more. Therefore, the end devices in an IoT system should be environmentally rugged and tamper proof to protect against any malicious forging

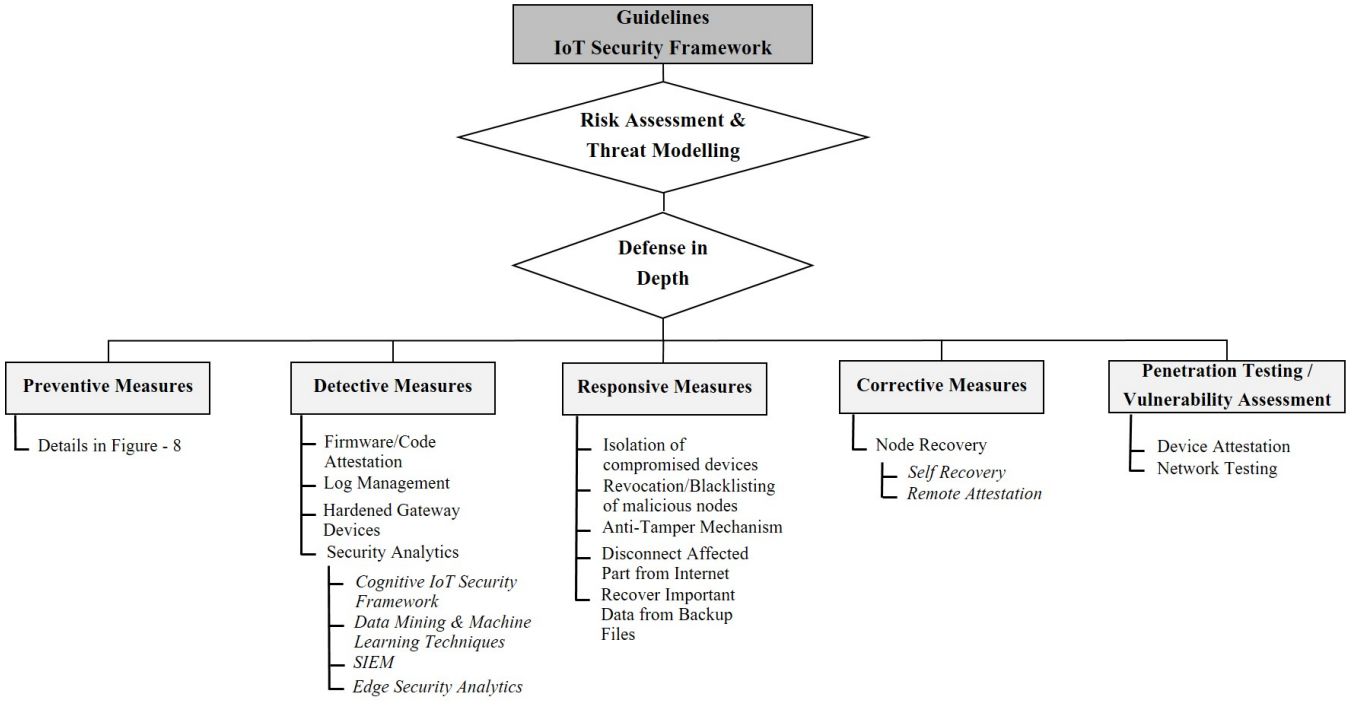

Fig. 13: Guidelines for IoT Security Framework

and access to device hardware. However, in case an adversary tries to tamper the device hardware physically, it should fail safely [157]. Such that upon detection of any tampering attempt, the device memory should automatically wipe off all the data it is storing. This may protect against illegal access to sensitive corporate data, cryptographic primitives (passwords, keys, unique identifiers of neighboring nodes etc.) or any intellectual property. Some of the embedded systems manufacturers implement end-to-end security in their devices, such as ARM mbed [158] provides secure boot and built-in cryptographic and protocol support to ensure secure network connection. Whereas, Juniper Networks [159] make use of Integrity Measurement Architecture (IMA)/Extended Verification Module (EVM) to detect any accidental and malicious file modifications. The files are attested before they are accessed. The attestation can be done locally or via remote attestation. NXP is yet another manufacturer and developer of various solutions for embedded systems [160]. It offers a secure authentication and anti-counterfeiting technology in the form of tamper-resistant CPU and a secure memory that can store cryptographic keys and a device identifier.

Given dynamic threat spectrum, the firmware of IoT devices also continuously evolve by installing periodic security and other operational updates. Therefore, it is imperative that all the IoT users especially the critical infrastructure owners such as smart grid, ICS, traffic control systems, nuclear power plants, air travel and railway systems, keep the software/firmware of their devices up to date to protect against any security vulnerability identified by the device manufacturers. Another important aspect of any distributed IoT system based on heterogeneous devices is authenticated and secure broadcast of security updates and control messages.

Change of default device configuration especially the security settings such as username and passwords should be implemented immediately upon the first installation of the IoT devices. In today's hostile environment security should not be an optional feature instead it should be implemented by design [157]. Hence, the device firmware should prompt the user for a change of default security settings before it starts functioning.

Data Security. Security of data mostly refers to the triad of information security, i.e., confidentiality, integrity, and availability of data. To ensure security of data, organizations must classify their data as per its sensitivity and then grant access to users according to their authorization to access respective class of data [17]. Moreover, in the current era of IoT, the privacy of data must not be ignored such that personal information should not be disclosed

publicly or to an entity not authorized to view. In the age of data-driven business development policies, security of PII (Personally Identifiable Information) in medical and financial records require due consideration. IoT business owners or cloud service providers should continuously weigh the utility of user data they are maintaining against the risk of holding it. Whenever the said ratio gets out of proportion, i.e., the risk of keeping large privacy-sensitive user data is more than its further utility; the companies should delete old data. Authors in [157] state that in case of corporate sector data theft, the unauthorized disclosure of intellectual property may create conflicts in ownership of such data. To ensure the security of private data, researchers in [15] suggest the use of ephemeral and separate identifiers during communications and while in storage.

In a cloud environment, there should be a secure device-to-cloud interaction. In a similar effort, IBM Watson IoT uses TLS v1.2 for authenticated and encrypted IoT device interactions, which ensure secure exchange of data over the network. The data sent from the end device to the cloud must be encrypted to preserve the confidentiality of user information [17]. However, the encryption of user data restricts intra-cloud processing and data analytics. To overcome such an issue, use of homomorphic encryption is recommended [161]. Authors in [17] also suggest the use of a Cloud Access Security Broker (CASB) that not only helps in maintaining a secure link between corporate network and the cloud services provider but also gives organizations insight into cloud applications and services being used by its employees.

Irrespective of the type of storage, data availability to authorized users is a critical requirement for any organization. Moreover, in the wake of a recent surge in ransomware attacks, security of relevant personal/corporate data is equally vital. It is recommended that a distributed storage mechanism should be preferred over a centralized storage to avoid a single point of failure. Blockchain provides a secure, unforgeable and a transparent mechanism for distributed storage, in which every transaction is validated by network consensus [162]. IBM Blockchain [163], Microsoft Azure [164] and Hyperledger Fabric by Linux Foundation [165] are few examples of multi-purpose Blockchain platforms.

Authentication and Access Control. Authentication for controlled access to an IoT system is not limited to devices only. Same applies to applications and gateway devices as well [17, 154]. It is required to protect sensitive information against malicious applications downloaded by the users from unauthorized sources. Similarly, gateway devices are to be authenticated to protect against the introduction of a forged gateway device in the network. Depending upon desired security level, multi-factor authentication may be used, i.e., a combination of password/passkey and a biometric identifier. Moreover, mutual authentication between IoT devices and IoT services/devices can prevent against masquerading of IoT services by malicious parties. In addition, it can further help in accountability and forensic analysis.

Considering the importance of network access control, authors in [166] proposed a traffic flow based network access control. It implements the access control based on numerous traffic flow identifiers, such as MAC address, source and destination address (IP address). Similarly, IBM Watson IoT uses IBM Bluemix that implements role-based controls for users, applications, and gateways to realize security of data and access to other services/resources [86]. Such a distinction between roles helps in the implementation of unified security policies across the complete network. In addition to role, geographical location [167], department, device type, OS/firmware version and the time of the day at which user seeks access [17] can also form the basis of access control policies.

Correspondingly, authors in [168] propose an identity-based cryptographic authentication scheme without the need of a Key-escrow mechanism to secure M2M interactions in CPS. The scheme saves upon precious computation and communication resources by averting the process of signature generation, transmission and verification. The proposed scheme is also claimed to be robust against MITM, impersonation, replay, DoS and node compromise attacks. In a similar endeavor, security researchers in [169] have designed a novel mutual authentication and key establishment scheme to secure M2M communication in 6LoWPAN networks. The proposed scheme duly cater for the static as well as the mobile nodes in a 6LoWPAN network. Respectively, [170] suggests a certificate-less anonymous authentication scheme based on hybrid encryption to secure multi-domain M2M communication in CPS. The proposed solution is considered to be tolerant against MITM, replay, impersonation, DoS, and node compromise attacks.

Controlled access to user data by third parties is an

important issue. Currently, user data owned by most of the online services is made available to the third parties in the form of APIs. The possibility of an unauthorized entity besides the generator of the information and the host service accessing the user information cannot be ruled out. Such an event can result in various privacy and ethical problems. Hence, authors in [171] propose an OAuth-based external authorization service for IoT scenarios. Instead of smart objects/devices storing the authorization related information and performing the computation intensive verification process, the verification of a request by a service is delegated to an external OAuth-based authorization service. Such an arrangement provides flexibility to the service provider (hosting user data) to remotely configure the access control policies. However, the delegation of authorization logic to an external service demands strong trust between the service provider/smart object and IoT-OAS (OAuthorization-based Authorization Service). There is also a requirement of a secure communication link between the service provider/smart object and IoT-OAS. Moreover, if the smart object directly offers its data as a service, then there is a likelihood of a DoS attack if the smart object receives a large number of simultaneous requests. The proposed scheme is also vulnerable to a MITM attack if the attacker uses an untrusted HTTP/CoAP proxy. In this way, an attacker can have access to the communication between endpoints and can also get hold of the authorization information. Based on the apprehended authorization information attacker can spoof the service requester's ID. The scheme also does not protect against a physical compromise of the device.

In another work, to facilitate and securely manage remote access by users to corporate networks/sites, [17] recommends software-defined perimeter to restrict access to legitimate users. In addition to mere user authentication, such a security perimeter ensures that the user accesses the applications, services and data as per his authorization only.

Software Integrity. It is to be made sure that only legitimate software is running on IoT devices, during initial bootup, at runtime and during firmware updates. Software integrity is one of the important pillars in IoT security as cryptographic algorithms, network security protocols, secure storage and other such tasks are implemented by software [43].

Mobile Applications. It is being covered as a separate entity because downloading of mobile applications from unauthorized stores is one of the primary sources of corporate networks infection. The organizations are advised to enable installation of only whitelisted apps on corporate devices and should provide a list of the same to its employees for implementation on their personal devices as well [17].

Security of Non-Corporate Smart Devices. Increase in use of smartphones, wearable smart devices such as fitness trackers/bands, smart watches and smart home appliances including smart thermostat, intelligent lighting system, smart TV, smart cooling system, smart doors, etc., has added another dimension to IoT ecosystem. It is a common belief that mobile phones, wearable or smart home devices do not contain sensitive information, so they do not require security [17]. Resultantly, manufacturers do not pay much heed towards security of these devices [11]. Due to this lack of security consciousness, IoT devices have recently been subjected to massive DDoS attacks [44]. It is also viewed that in future, nation states can sponsor the sale of apparently legitimate IoT devices for cyber espionage [17] or sabotage of target systems. Therefore, it is recommended that a minimum security standard should be set for mobile/wearable smart devices with an emphasis on following: Access to device based on at least two factor authentication, i.e., password and a biometric identifier, limited access to corporate data (only viewing option without any modification rights), storage of sensitive data such as health and financial information in encrypted form.

The corporate networks should provide remote access to those devices only that meet the minimum security requirements. It is also recommended that enterprises should enable mobile access to their systems through VPNs based on multi-factor authentication.

Key Management. Secure key management is the baseline for the security of any IoT system. It includes key generation, key distribution, key storage, key revocation and key updates. TCG provides a hardware-based secure key management system that supports various options for provisioning of keys during IoT device lifecycle, i.e., during chip manufacturing, assembly of the device, while enrolling with a management service and during owner-personalization. It also provides secure key update over an untrusted network [156].

Network Segmentation. Network segmentation or segregation is an effective methodology to curtail the impact

of a node or a part of network compromised by an adversary. It not only protects networks and systems of different security classifications but also protects systems of the same classification with varying security requirements. Depending on the system architecture and configuration, network segmentation can be achieved by various methods. Some of these include implementation of demilitarized zones, physical isolation, use of VLANs, software-defined perimeter, application firewalls, application and service proxies, user and service authentication and authorisation, and last but not the least content-based filtering [172].

Virtualized Security. The shift from hardware to Software Defined Networks (SDN) has revitalized the flexibility in the implementation of effective security measures. Virtualized security has enabled protection of data irrespective of its location. Another benefit of this virtualization is that instead of maintaining dedicated hardware for numerous security protocols such as encryption, secure routing, and secure gateways, software-based security solutions can be implemented on a single shared platform. Such a dynamic security solution will enable organizations to enforce security policies with persistence in every type of IoT system, i.e., private or cloud-based IoT architecture.

An example of SDN-based security enhancement for IoT systems has been demonstrated in [25]. The researchers believe that SDN can be used to augment IoT device-level protections by implementing dynamic security rules at the network level. To achieve this goal, researchers in [25] have proposed a software-based Security Management Provider (SMP) that provides appropriate access control functionality to the users of IoT systems such as smart lighting, smoke alarm and baby monitor, to preserve their privacy and further improve the security. SMP exercises dynamic configuration control over ISP network and the home router on behalf of the user. It communicates with the ISP network via APIs and also interacts with the IoT system users via GUIs. The proposed security solution thus motivates the manufacturers to concentrate less on UI (User Interface) development and instead focus on the development of APIs that allow a third-party, i.e., SMP to configure IoT behaviour at various layers of IoT architecture.

In yet another work, [173] proposes an SDN-based security architecture for heterogeneous IoT devices in an Ad-Hoc network. The proposed architecture comprises smart nodes, OpenFlow enabled nodes, OpenFlow enabled switches and distributed SDN controllers. The multiple SDN controllers are synchronized to provide a granular network access control and network monitoring. Hence, all network devices are first authenticated by the controllers, before they start accessing network services as per their authorization.

Conclusively, it is the SDN controller that monitors and manages all aspects of the network including security, and the interface between SDN applications and the hardware components [174]. Hence, SDN controller, being a focal point of all the control activities can be termed as a lucrative target for the malicious attacks. Thereby, a successful attacker may gain unauthorized access to the controller and insert viruses or malware in the network thus threatening the confidentiality, integrity and the availability of data and other network services [57]. Similarly, authors in [174] also identify various threats to SDN such as unauthorized access, data leakage, data modification and misconfiguration. The authors also highlight the eavesdropping and jamming threats on the physical layer of Software Defined Optical Networks (SDON). However, they also underline a security measure to protect against eavesdropping and jamming in optical lightpath based on a hopping mechanism. But such a mechanism also suffers some shortcomings concerning secure exchange of hopping sequence between the transmitter and the receiver and protection against MITM attacks. It is, therefore, imperative to protect SDN against such single point of failures and attacks on centralized controllers.

Adaptive Security Management. Most of the IoT applications such as eHealth monitoring comprising BSN with dynamic network topology, require adaptive security management. Authors in [41] propose a metrics-driven adaptive security management model for eHealth IoT applications. The proposed security model monitors and collects the security contextual information from within the system as well as from the environment. Based on collected data, it measures the security level and matrics, analyzes the received data and responds by changing the security parameters such as encryption scheme, authorization level, authentication protocol, level of QoS available to various applications and reconfiguration of the protection mechanism.

Security of Automated M-2-M Communication. In an IoT ecosystem, M-2-M communication is an important pedestal of industrial and critical infrastructure

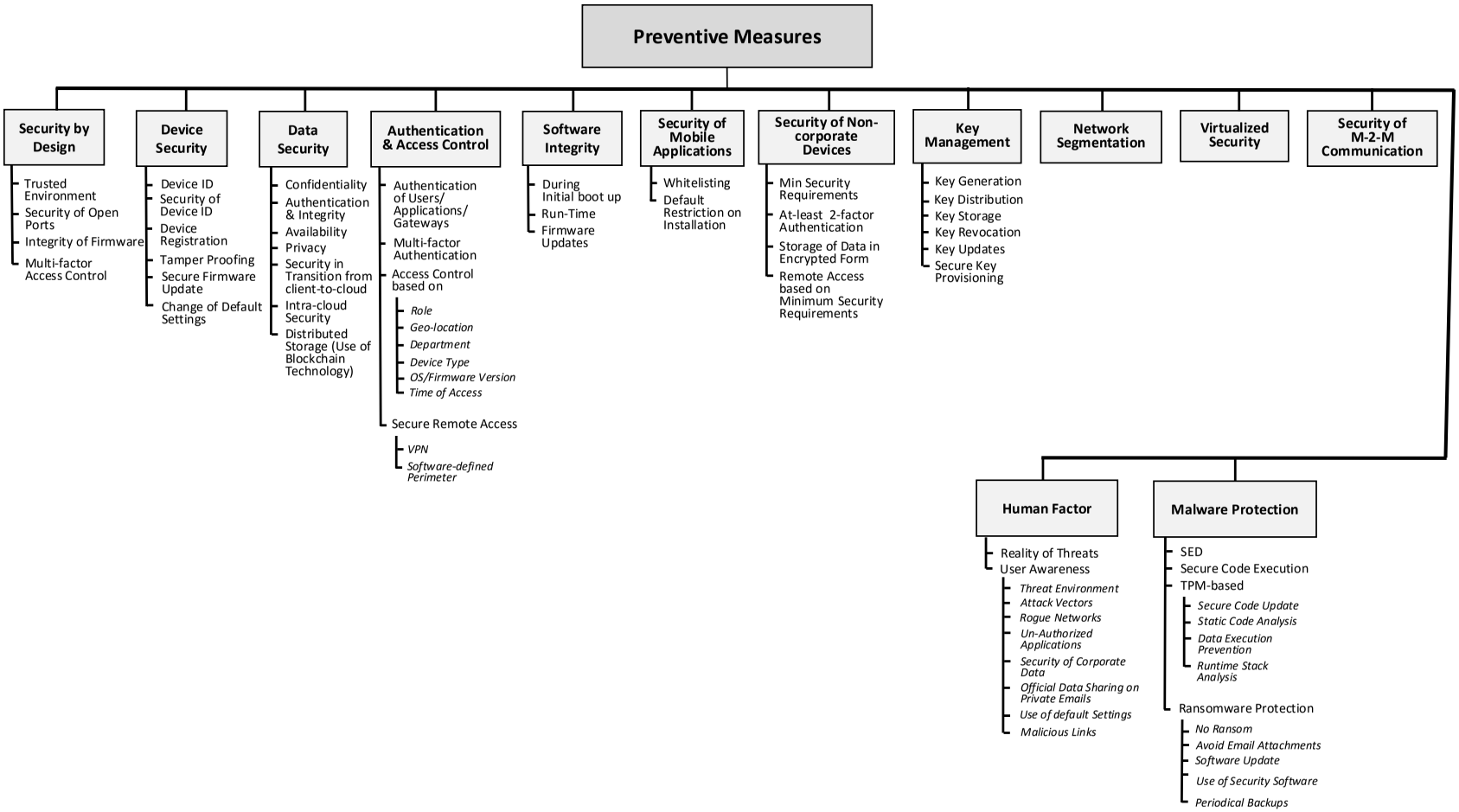

Fig. 14: Guidelines for IoT Security Framework - Preventive Measures

TABLE V – *Security Measures and Their Impact*

Ser	Security Measure	Impact / Threat Protected Against	References
1.	Risk assessment and threat modelling	Identification of all possible threats, vulnerabilities and risks. Helps in the development of a risk mitigation plan and formulation of a composite security framework	ISO-27001 [150], NIST Special Publication 800-30 [151], Cisco [152]
Preventive Measures			
2.	Security by design from the vendors (Change of default security settings on device startup, security of all debug ports/interfaces)	Users' unawareness, unauthorized access to the devices through backdoors, firmware and software modification	AT&T [17], IBM [153]
3.	Device identity (ID) management	ID spoofing and device replication attacks. Compliments network security protocols (IPSec, TLS, SSH)	IBM [154], TCG [155, 156]
4.	Tamper-proofing of IoT devices	Unauthorized disclosure of cryptographic keys and passwords, modification of code/firmware and replication/cloning of devices	IBM [154], NXP [160]
5.	IoT device registration and management	Unauthorized or illegal device joining the network	IBM [43]
6.	Secure boot and builtin cryptographic protocols support	Unauthorized access to device and modification of the boot sequence to execute malicious codes	ARM Mbed [158]
7.	Use of Integrity Measurement Architecture (IMA) or Extended Verification Module (EVM)	Accidental and malicious modifications of files	Juniper Networks [159]
8.	Data classification and requisite user authorization	Unauthorized disclosure and access to data	AT&T [17]
9.	Use of ephemeral identifiers for communication and storage of data	User privacy in the context of PII	IBM [15]
10.	Identity-based authenticated encryption and mutual authentication schemes for CPS,	Impersonation, MITM, eavesdropping, data forgery, replay and modification attacks	[168, 169, 170]
11.	Homomorphic encryption	Privacy issues in cloud-based IoT during data processing/analytics	[161]
12.	Cloud Access Security Broker (CASB)	Security issues in cloud-based IoT systems	AT&T [17]
13.	Blockchain Technology	Data integrity issues including data modification and forgery, replay attacks, malware attacks targeting data security, integrity and availability such as cryplocker, ransomware and wiper	Bitcoin Blockchain [162], IBM Blockchain [163], Microsoft Azure [164] and Hyperledger by Linux Foundation [165]
14.	Authentication and access control in applications (including white/black listing)	Downloading of malicious applications	IBM [154]
15.	Endpoint and gateway device authentication and access control	Introduction of forged end/gateway devices in the network by an attacker	IBM Bluemix IBM [86]
16.	Authentication between devices within an IoT system	Masquerading of IoT services by malicious parties. It also facilitates accountability and forensic analysis	

Continued on next page

TABLE V – *Continued from the previous page*

Ser	Security Measure	Impact / Threat Protected Against	References
17.	Role-based access control for the users of an IoT system (In addition to role, access control policy can also consider geo location, department, device type, OS/firmware version and time of the day)	Security and privacy issues related to data and unauthorized access to the network services	IBM [86], Cisco [167]
18.	Ensure software integrity during initial boot up, at runtime and during firmware/software updates	Code modification and malicious code execution	IBM [43]
19.	Security of data in personal IoT devices (Smart watch, smartphone, health monitor, fitness tracker) by using lightweight cryptographic protocols	Unauthorized access/disclosure to personal information	
20.	Secure remote access to corporate networks from smart IoT end-devices using VPN and limiting access to end-devices meeting minimum security standards	Attacks on corporate networks, security issues related to business data/intelligence	US-CERT [142]
21.	Key management (including key generation / distribution / storage / revocation / updates	Masquerading attacks and device compromise	
22.	Network segmentation using (Demilitarized zones, physical isolation, VLANs, software defined perimeter, application firewalls/proxies and content-based filtering)	Curtail impact of a node or a part of network compromise	Australian Signals Directorate [172]
23.	Virtualized security based on SDN	Augment IoT device-level protection by implementing security at the network level. Hence, reducing burden of cost related to the development of security protocols for low-cost IoT devices for the manufacturers	[25, 173]
24.	Use of self-encrypting devices/drives (SED)	Unauthorized disclosure of data	TCG [155, 175]
25.	Adaptive security management	Provides dynamic re-configuration of security parameters	[41]
26.	Execution of signed binaries, TPM-based secure software updates, static code analysis, runtime stack analysis	Malware attacks	TCG [155]
27.	Runtime restart of RT-IoT devices with tight timing constraints	Malware attacks	[176]
28.	Security awareness workshops and lectures for the employees	Social engineering attacks, phishing/spear-phishing attacks, download of infected/malicious apps	
Detective Measures			
29.	Runtime verification of firmware/code	Malicious code, corrupt software	
30.	Log management	Facilitates detection of security breaches	
31.	Network security analytics	Detects security breaches, malfunctions and anomalies	Cisco [167], IBM-CIoT [177, 178, 179]
32.	Edge security analytics	Facilitates isolation of security events at the source and limit attack spectrum	IBM [43]

Continued on next page

TABLE V – *Continued from the previous page*

Ser	Security Measure	Impact / Threat Protected Against	References
33.	Network level security measures to enforce cross-device security policies	Manipulation of actuator actions based on malicious/modifies sensors data	[38]
34.	Penetration testing and vulnerability assessment	Detect/identify weaknesses in all layers of IoT protocol architecture to facilitate respective countermeasures	
Responsive Measures			
35.	Incident response plan	To streamline the response in case of a security incident and facilitate in recovering from the attack by adopting requisite corrective measures	
Corrective Measures			
36.	Self-recovery and diagnostics, and remote attestation	To recover from the security incident by reconfiguring the devices and removing all remnants of the attack	TCG [155]
37.	Secure reboot of RT-IoT devices	To recover from malware that resides in the RAM	[176]

automation such as power plants, intelligent traffic control system, railways, smart grids, and smart cities. This type of communication ranges from information sharing between robotic/intelligent controllers and smart actuators/appliances to data sharing between smart vehicles. The automated exchange of information between unknown entities must meet the security and privacy requirements. Taking the example of IoV (Internet of Vehicles), it is recommended that any proposed solution should meet specific security requirements such as data authentication, data integrity, data confidentiality, access control based on authorization, non-repudiation, availability of the best possible communication link and anti-jamming measures [180].

Protection Against Malware Attacks. There is an increasing trend in ransomware attacks over the last four years in which the number of attacks has risen to 638 Million in 2016 from 3.8 Million in 2015 [181] and are still being counted in 2017-18. As per Symantec Corporation [182], ransomware attacks increased by 4500% in 2014, being too profitable for cybercriminals. Symantec Corporation has proposed few dos and don'ts for the consumers and businesses to protect themselves from such attacks. The preventive measures include: Do not pay the ransom, avoid clicking attachments in unknown emails, keep software up to date, must use security applications and finally the most important step is to take periodical backup of valuable data.

Some common security measures against most of the malware attacks include, not to use hardwired/default username and passwords. In addition, use only authenticated and encrypted protocols for inbound connections, i.e., SSH (Secure Shell) for telnet, SFTP (Secure File Transfer Protocol) for FTP (File Transfer Protocol) and https for http. Finally, keep all external interfaces of the administrative connections closed. Security at lower layers should be complemented by application level access control, use of multi-factor authentication protocols, use of OPC tunnelling technologies, installation of update patches, deployment of software restriction policy (application white-listing), white-listing of legitimate executable directories, use of IPSec or VPN for remote access [142], implementation of ingress and egress filtering, restricted number of entry points to ICS Network, maintenance of logs and use of configuration management tools to detect changes on field devices.

Similarly, numerous security solutions proposed by TCG technologies [155] help to prevent unauthorized access to security-critical programs and data. To solve this issue Self Encrypting Drives (SED) based on TCG specifications are in common use for embedded systems such as ATMs, secure mobile phones, corporate copiers, and printers. In these drives, encryption is implemented in the hardware, and data is automatically encrypted in a transparent way to the user. The drives can be safely sanitized for reuse without any need for rewriting multiple layers of garbage data. The user is just required to delete the cryptographic key. As a result, the data stored is made illegible. The hardware-based automatic encryption is termed efficient and secure than simple software-based encryption, which can be turned off anytime by the user

[175].

In addition to restricting unauthorized disclosure to sensitive data, the malware should be prevented from execution from the beginning. The two best techniques for this purpose are whitelisting and execution of manufacturers' signed binaries only. TCG offers TPM-based secure software updates, static code analysis, data execution prevention and runtime stack analysis. Any combination of such techniques can ensure the integrity of a runtime environment [155]. Although hardware-based security protections are always efficient and more secure than software-based solutions, however keeping in view the cost effect and hardware complexity, these techniques may not be feasible for resource-constrained embedded devices such as wireless sensors and actuators. In such cases, the best way is to program the device to reboot periodically and make use of boot time protections. However, rebooting a sensor or actuator periodically may degrade the performance of resource-constrained devices. Such devices are usually battery operated and have limited energy. Hence, frequent restarts may drain the device's resources. Another, limitation of restart-based recovery mechanism is that it can destabilize RT-IoT (Real-time IoT) systems that need consistent actuation with tight timing constraints. To address this issue, authors in [176] propose a runtime restart-based security protocol "ReSecure" for RTS (Real-time Systems). ReSecure is a blend of hardware and software mechanisms that enable a tradeoff between the security guarantees and control performance while ensuring the safety of the physical system at all times.

Human Factor. Any level of security is not sufficient until the users of the respective organization are security concious and believe in the reality of the threats. Any unintended action like connecting an infected USB flash drive to a company's private network can cause a disaster for that enterprise. The organizations should deploy network-wide security policies to implement controls based on authentication, authorization, role and even incorporating geolocation of the users. Enterprises should organize periodic security updates and awareness lectures for its employees covering following dimensions:

- Current threat environment.
- Attack vectors being used by hackers/adversaries.
- Implications of sharing sensitive corporate and personal information on public/rogue networks.
- Downloading and installation of applications/software from unauthorized sources.
- Storing of corporate data in personal laptops/flash storage devices that too without encryption.
- Use of private email accounts for official purpose.
- Throwing of important official documents in open bins, thus giving an invitation to the attackers for dumpster diving.
- Use of default settings for smart devices.
- Sharing of sensitive data over social media that too with default (lowest) security settings.
- Avoid malicious links in unknown emails.

4) ***Detective Measures****:*

Firmware/Code Attestation. Runtime verification of firmware/code installed on an IoT device is an important means of detecting execution of a malicious code installed remotely on a device.

Auditing (Log management). A record of all changes made to the system and devices be maintained to enable periodic audits to detect security breaches.

Hardened Gateway Devices. Security hardened gateway devices can be used to monitor sensors data feed to determine the health of communication b/w devices and services-based applications.

Security Analytics. It helps in gaining visibility of the IoT ecosystem and ultimately controlling all the network components including the hardware and software to detect and rectify any malfunction or a threat [167]. IBM uses a Cognitive IoT (CIoT) Security Framework named Security-360. All the network components including devices, users, applications, business processes and even workload contribute to form a 360-degree view of the security posture. Based on data provided by the entire environment, the security mechanism assesses the changes in the security posture of the network and plans a defense. In this regard, various data mining and machine learning techniques can provide automated methods to track normal behaviour and flag anomalies [177, 178, 179]. Moreover, Security Information and Event Management (SIEM) is also considered a vital component of a defense-in-depth approach to network security. It is therefore concluded that intelligent threat analytics should be able to protect the IoT ecosystem against all sort of threats based on known signatures, predictable malicious behavior [17] and correlation of security incidents/events.

A subset of overall system security analytics is "Edge Se-

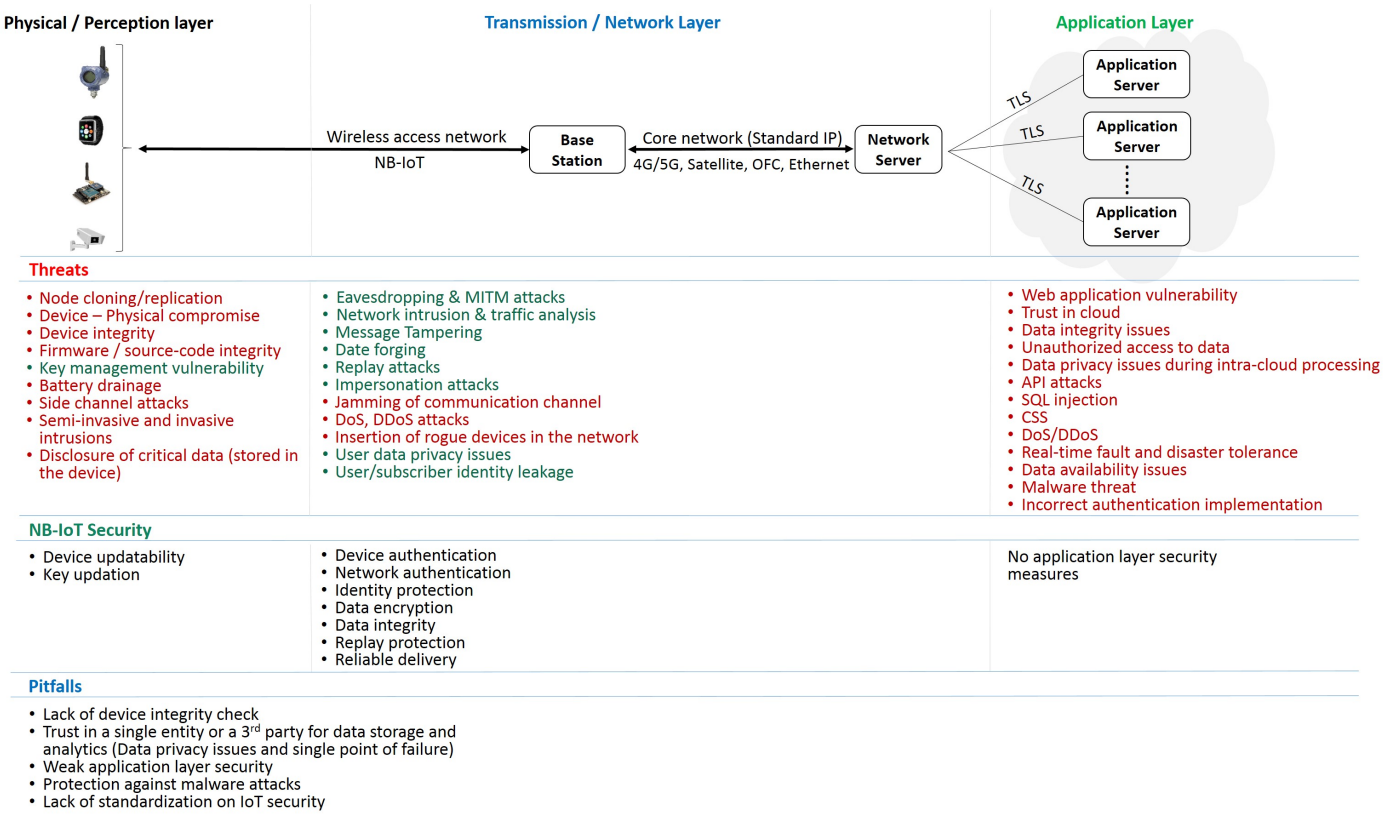

Fig. 15: NB-IoT Security in IoT Threat Environment

curity Analytics". It is implemented by deploying security intelligence gateways. These intelligent devices provide swift responses to security incidents by faster detection of anomalies and re-mediation by isolation of events at the source and limiting attack spectrum. They also help in preserving the privacy of sensitive data by carrying out processing locally [43].

Redefining Network Level Security for IoT. Today, IoT device manufacturers just focus on novel functionality, easy operation and earliest product launch in the market. Hence, they do not give attention to device security. This lack of manufacturers' attention to security coupled with constraint resources, IoT devices are not suitable for traditional host-based protections (anti-virus and security patches). Hence, researchers in [38] proposed a network level security architecture to secure IoT devices. Their security architecture employs an IoTSec (security controller), µmboxes (gateways for IoT devices) and IoT end nodes.

The IoTSec controller centrally monitors the network to record security contexts and environmental variables for each end device, to form a global view of a set of possible states of the system. Based on the set of states IoTSec decides or controls the flow of commands to the end devices. The proposed system is claimed to be equally useful to enforce cross-device security policies. E.g., in a smart home, if an attacker hacks into a fireplace and commands it to ignite the fire in order to cause an accident. To address this vulnerability, the IoTSec controller ensures that the fireplace is turned on only if the camera detects that someone is present in that room. The status of camera output, i.e., the presence of a person in the room can be read from the current global state of the smart home maintained by the IoTSec controller itself. However, certain issues related to centralization of IoTSec controller and the limitation of using different µmboxes for every other kind of IoT device needs to be addressed.

5) ***Responsive Measures****:* An effective incident response plan begins even before any security incident occurs. In an IT environment, the response team is usually called as Computer Emergency Response Team (CERT). These teams comprise skilled cybersecurity professionals, auditors, legal experts, IT administrators and other specialized members. The goal of CERT is to develop

and physically practice a comprehensive response plan against any security breach so that all the stakeholders are clear about their responsibilities. An organized and well-planned incident response can make or break any business. The response measures are also termed as after-incident reactive measures, which include:

- Action against compromised devices/parts of the system allowing rest of the system to run its routine functionality.
- Revocation and blacklisting of malicious nodes.
- Initiation of anti-tamper mechanism, in which, as soon as the hardware of the node is interfered with, the node's memory containing firmware and the code should immediately be wiped off, and the node should only join the network after being activated by personalization instead of OTAA (Over The Air Activation).
- Disconnect connectivity from the internet.
- Isolation of compromised sub-systems so that healthy part of the network remains available.
- Recover important official and personal data from backup.

6) ***Corrective Measures****:* Once an IoT device is compromised, detected and isolated from the network, the next step is node recovery, i.e., secure firmware/code update and reactivation of the device. There are two methods of node recovery. The first one is self-recovery, in which, the device itself performs the integrity check of the code running on it and the last best configuration stored in read-only storage. If the validation fails, the device deletes the current code and reinstalls last best configuration. The device then restarts and performs validation of all its modules. The second method is remote attestation; the device sends integrity report to the controller/gateway device for remote validation [155]. A secure firmware update process is then initiated by the verifier if the validation fails.

7) ***Penetration Testing/Vulnerability Assessment****:*

Device Attestation. Periodic device-side code analysis should be performed to check for the presence of any malicious code or modification in the original code. The successful code verification helps in shrinking the attack surface [15].

Network Testing. Use of penetration testing toolkits and other vulnerability assessment measures adopted by ethical hackers to secure the network. The most common tools are Metasploit, Wireshark, Nmap, Social Engineering Toolkit, Kali Linux, Nessus, etc. The penetration testing is done to highlight the weaknesses in the target system. The testing can be performed on networks, websites, and servers. The weaknesses are then fixed by installing security patches, improving security configurations, making changes in the IDS and firewall rules and security of open ports/interfaces.

8) ***Cost-Benefit Analysis for the Selection of Suitable Security Measure****:* In this section, we have presented a defense-in-depth approach for IoT, comprising various preventive, detective, and responsive measures. Here a question arises that what about the complexity and cost comparison of various security measures? In response to this question, authors in [183] illustrate that the security requirements of two distinct IoT systems and even the security features of two different technologies cannot be compared using a single measure. The security measures are adopted as per the technical resources (computational power, battery life, memory and available bandwidth) of end devices, and the threat environment. However, some traditional host-based security solutions such as anti-virus, frequent security updates/patches, secure execution environment, OS virtualization, etc., are difficult to be implemented on resource constraint IoT devices. Hence, a relative cost-benefit analysis of security measures providing same level of security is essential, to select the suitable technology. E.g., as discussed in Section-IV.A.(3), allocation of a unique device identifier is essential to protect against ID spoofing and device replication attacks. However, just allocation of an identifier is not enough, the safe storage of device identity and other associated cryptographic primitives such as private keys and symmetric keys require additional measures such as TPM-based keys [155, 156]. However, any additional security measure comes at the cost of additional overheads in the form of special hardware, high computation and energy costs etc.

Similarly, blockchain, a distributed ledger technology, is recommended to replace centralized cloud platforms. Both blockchain and cloud store data for further processing. Both technologies ensure data authentication and integrity. But there are few differences that play a key role in the selection of a suitable technology for IoT. Cloud services are provided under the centralized control of one trusted entity. Hence, the cloud is vulnerable to

TABLE VI:
Comparison of LPWA Technologies

Feature	LTE-M	NB-IoT	LoRaWAN	Sigfox
Licensed spectrum	Yes	Yes	No	No
Device / subscriber authentication	UICC/eUICC	UICC/eUICC	Yes	Device only
Network authentication	Yes LTE-AKA	Yes LTE-AKA	Optional	No
Identity protection	TMSI	TMSI	Partial	No
Data confidentiality	128-AES	128-AES	Yes (AppSKey)	No
Data integrity	Limited	DoNAS (Optional)	Yes	Yes
Control signal integrity	Yes	Yes	Yes	Not known
End-to-Middle security	No	No	Yes	No
Forward secrecy	No	No	No	No
Replay protection	Yes	Yes (Optional)	Yes	Yes
Reliable delivery	Yes	Yes	No	No
Device updatability	Yes	Yes	Limited	No
Keys updatability	Yes (Optional)	Yes (Optional)	Limited	No
Updation of long term keys	Yes (OTA)	Yes (OTA)	Limited	No
Requirement of certified equipment	Yes	Yes	Optional	Yes
IP network	Yes (Optional)	Yes (Optional)	No	No

the single point of failure concerning security and privacy issues [184] including data manipulation [185, 186], and the availability of cloud services. Concerning data manipulation, the cloud service provider has to be the trusted party as it has control over the data stored in the cloud and related services. Therefore, the cloud provider can manipulate user data [186]. Whereas, blockchain is orchestrated in a way that all the miner and full nodes in the blockchain network maintain a same copy of the blockchain state and the trust is distributed among all the network nodes. Hence, if one device's blockchain data is altered, the system will reject it, and the blockchain state will remain un-tampered. Correspondingly, single point of failure also concerns the availability of the services when the cloud servers are down because of software bugs, cyber-attacks, power problems, cooling and other issues; users find it difficult to access the cloud services [185]. Whereas, in the blockchain, data is replicated on many computers/nodes and problems with few nodes do not disrupt the blockchain services. Cloud is also vulnerable to un-authorized data sharing. E.g., in the recent past, private data of 87 million users was provided by Facebook to a British political consulting firm “Cambridge Analytica” without users' permission [187, 188]. Such a data breach results in irreversible data security and privacy issues. Whereas, blockchain with its smart contract technology gives users the freedom to restrict access to their data to authorized entities only, without placing trust in any third party or a cloud service provider [189].

Currently, blockchain is considered to be computational and energy intensive in the back drop of PoW-based consensus protocol used in Bitcoin Blockchain. However, considerable research is being done to design and develop IoT-specific blockchain technologies that infer low computational and energy costs [190, 191, 192, 193], are scalable [194, 195] and also offer privacy-preserving computations on user data [196]. Hence, it is the cost benefit analysis, the resourcefulness of end devices, and security requirements that holistically determine an appropriate security framework for an IoT system/use case.

V. Summary, Lessons Learnt and Pitfalls

To reach some logical conclusions/lessons and identify pitfalls concerning IoT security, we have projected a snapshot of the impact of security provided by one of the selected real-world IoT technologies on IoT threats discussed in this paper, in Figure-15. Although, there are

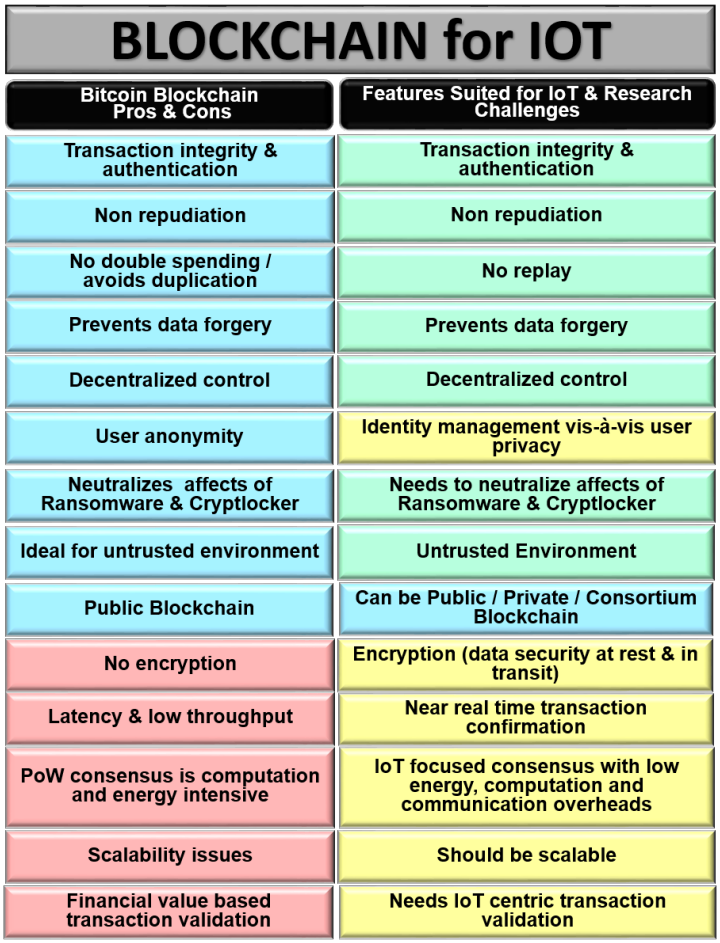

Fig. 16: Blockchain for IoT

many IoT communication technologies such as Zigbee, BLE, RFID, LTE-M, LoRaWAN, etc., that connect IoT devices with the gateways or base stations. However, LPWA (Low Power Wide Area) is considered to be a suitable technology for many IoT use cases due to its low power consumption, wide coverage, long range, low latency, reliability, low cost, better QoS, and considerable security [50, 183, 197]. Therefore, we have carried out a comparison of various LPWA technologies in Table-VI. As shown, there are various options for LPWA technology in both licensed and unlicensed spectrum with varying security features. However, all of the technologies cannot be discussed here in detail. Therefore, we have only mapped NB-IoT security features in Figure-15. Under the threats sub-section of the Figure-15, the points shown in red color are the threats/attacks that are not protected against by the NB-IoT security features. Whereas, the points shown in the green color are addressed by NB-IoT. It is evident that NB-IoT protects against the majority of the transmission/network layer attacks and only a few perception layer threats. Moreover, the application layer threats make it essential for the application developers to embed requisite security measures in the applications. It is evident from Figure-15 that the cryptographic security provided by the NB-IoT, cannot protect against device capture and device tampering. Moreover, there is also no mechanism to detect any forging or change in the device code, hardware configuration, and system files. Such a protection is critical to detect remote code execution attacks that covert the devices into bots. The pitfalls observed in NB-IoT security are also shown in Figure-15.

As shown in Table-VI, LTE-M and NB-IoT operate in a licensed frequency band, whereas, LoRaWAN and Sigfox operate in an unlicensed spectrum [183]. Hence, it is imperative to discuss the impact of a licensed and an unlicensed frequency spectrum on the operational performance and security of an IoT system. The main advantage that NB-IoT has over LoRaWAn and Sigfox

is that being in a licensed frequency band, NB-IoT is based on an international standard defined by 3GPP [51]. Therefore, NB-IoT is mature with good QoS and is also less vulnerable to interference. Although, the cost of a licensed frequency band is very high, i.e., more than 500 Million USD per MHz, yet, the security and the performance benefits outweigh the cost effect. Being operating in a licensed spectrum the end devices get access to the network after due authentication and authorization only. Therefore, it is difficult for an attacker to introduce a forged device in the network. Moreover, a regulating authority can control and manage a licensed spectrum with much ease as compared to an unlicensed one.

On the other hand, LoRaWAN is a non-standard proprietary technology with low QoS and no message delivery reliability. Being in an unlicensed frequency band, LoRaWAN, and SigFox are at high risk of service degradation as the frequency band is shared with a lot of other radio devices. Moreover, the use of unlicensed spectrum in most countries is regulated with some restrictions on the service providers concerning maximum power of the transmitted signal and the duty cycle. However, still, it is difficult to control and regulate the unlicensed spectrum as at times there can be a large number of ad-hoc networks operating in the said band. Correspondingly, the limitation on the duty cycle makes it difficult to support firmware updates over the air [198]. Whereas, IoT devices without any software updates or security patches are a security hazard. The brief discussion on the impact of real-world IoT technologies on the security threats and the previous discussion on IoT threats and security framework has led us to draw certain lessons which further helped us to identify the pitfalls in the current IoT security environment.

Lessons Learnt and Pitfalls

- As shown in Table-II, IoT threats at various layers such as physical, MAC/Network and application layer exploit different vulnerabilities and use different attack vectors to achieve malicious objectives. E.g., a device manufacturer leaves some open interfaces in the device hardware. These open interfaces can be exploited by the attacker to gain an unauthorized access to the device and manipulate its operation [71]. Similarly, jamming of a communication channel targets availability of the network or network services. Whereas, anti-jamming protection requires different approach as compared to merely protecting against eavesdropping. Hence, attacks at various layers will have different impact on the overall security of an IoT system and will require different security measures depending upon the IoT use case and threat environment.
- According to the discussion in Section-II.C, attacks at physical layer such as device capture, jamming of wireless channel, hardware exploitation, node cloning, invasive intrusions, device configuration and firmware modification cannot be protected only by cryptographic security provided by IoT communication protocols. Therefore, security has to be viewed as a whole and supplementary measures need to be taken at different layers based on the security requirements of IoT use cases. These additional security measures may infer some additional costs in the form of hardware, software, bandwidth, computation or storage.
- The discussion in Section.II.D infers that depending upon the type and physical environment of IoT applications, end devices are vulnerable to physical attacks including device capture, tampering, invasive hardware attacks, side-channel attacks, reverse engineering, sensitive data leakage and firmware/source code modification attacks [42].
- DDoS attacks are mostly launched through compromised IoT devices [65]. Therefore, there is a requirement of an effective ingress as well as egress filtering, especially where IoT is connected to the internet.
- Cyber attacks are considered as one of the biggest threats to IoT applications [199], and mostly the network and the application layers are the focus of the attackers [199].
- No operation in an IoT system can be termed safe unless the integrity of the code installed on the IoT device and the integrity of the data being shared between devices is ensured [9].
- Absence of anti-virus/malware detection mechanism in IoT is one of the causes of successful attacks on the integrity of the code/software of an IoT end device [8, 9].
- Secure firmware update is one of the effective solutions against malware attacks in IoT. However, low downlink data rate, very short duty cycle and lack of firmware integrity verification measures make it hard for an IoT technology to implement an effective

firmware update mechanism [198].

- Not all IoT technologies' security protocols meet the needs of all IoT use cases. Instead, all technologies have adequate security for some specific applications. However, if the security provided is not enough for a particular use case, additional security measures can be taken but at the cost of some additional hardware, more computation or bandwidth cost, etc,.
- Security features of two different technologies cannot be compared using a single factor/measure.
- The ideal LPWA technologies have some important security features as optional. These features are required to be enabled by the network operators. Hence, the user organizations/network operators need to have a clear understanding of what security features they require for which IoT use case [183].
- To effectively provide comprehensive security and privacy solution, it is necessary to analyze the IoT application and associated threats. Although similar, a smart building is different from a smart work environment. The solutions, especially the ones involving classical cryptography and physical layer security must be tailored for the specific threats. The goal is to provide a cost-effective solution, while also taking into account the energy requirement of the various solutions (many devices can be battery-operated) [200].
- Mostly, security is not the primary concern while designing IoT technologies or products. Instead, the manufacturers focus more on the performance aspects such as low cost, low power consumption, more coverage, high data rate, ease of implementation and service delivery.
- Standard IT security protocols cannot be deployed on resource constraint IoT devices. However, selected standard security protocols can be optimized by removing various optional features.
- Security is a holistic property. Hence, it should not be considered in isolation. E.g., LPWA technologies are developed with the primary objective of improving upon the performance and reliability concerning low power consumption, wide coverage, long range, low latency, reliable data transmission, low cost, and better QoS security [50, 183, 197]. Therefore, some compromises have to be made between security and performance of the system. E.g., use of light weight cryptographic solutions to reduce the computation overhead and power consumption. Similarly, efficient use of available bandwidth implies the use of security measures with less communication complexity.
- Based on the discussion in Section II and Section IV.A on threats to IoT and guidelines for IoT security framework respectively, it is deduced that considerable research and development is being done in both academia and the corporate sectors to mitigate threats to IoT. These threats fall in the domain of security triad, i.e., threats to confidentiality, integrity, and availability of data/information. As highlighted in Section IV.A, that security has to be viewed as a whole, and for a defense-in-depth approach against IoT threats, we need to deploy various preventive, detective, responsive and corrective security measures. Hence, Table-V shows that there are many commercial off-the-shelf (COTS) and academic security solutions available/proposed to provide preventive, detective, responsive and corrective measures. For instance, issues concerning device security such as device identity [154, 155, 156], tamper-proofing [154, 160], registration and management [43], and secure boot [158] have been addressed by various tech giants including IBM, AT&T, TCG and Juniper Networks. Similarly, issues concerning data security and network access including authenticated encryption [168, 169, 170], privacy preserving computation (homomorphic encryption) [161], secure cloud access [17], mutual device and gateway authentication [86], and secure network access control [86, 167] have also been meticulously tackled.
  Whenever we talk about cryptographic security, key management is an associated challenge, and it is always considered to be an open research issue [200]. After, device, data, and network security, application layer security is also very essential as mostly the network and the application layers are the focus of the attackers [199]. Therefore, [89, 154] highlight threats to IoT systems that rely on websites and application for service delivery, and also propose security measures.
  However, the constrained resources in IoT devices and corresponding lack of strong security measures result in certain short comings that need to be addressed in future. These include; absence of an International IoT standards body that should govern minimal security standards as per sensitivity and nature of IoT application. Next is, the lack of security

mechanism to ensure the integrity of IoT devices. Similarly, protection of IoT devices against malware attacks and related secure firmware update are still open challenges. Another critical aspect is that, most of the data processing and analytics is performed under the centralized control of a third party/cloud provider that has to be a trusted one [95]. However, trust in a single party results in various security and privacy issues. Finally, more work is required to be done in intra-cloud and distributed privacy-preserving data analytics. Similarly, exploitation of zero-day vulnerabilities, especially at the application layer, is a persistent threat. Some of these vital open issues are discussed in detail in the next section.

VI. Open Research Challenges

A. Baseline Security Standards

Because of current lack of standardization on IoT products, diverse IoT applications and heterogeneity of IoT products, there are issues of security, interoperability and compatibility. Most of the IoT products are being manufactured without any baseline security standard [27]. Whereas, keeping in view the current threats, there is a requirement of various integrated security measures in IoT devices. These measures include requisite user authentication and authorization, encryption of data at rest and in transit, hardware security against tampering, and OS/application security. However, taking into account the constraint resources of many IoT devices such as sensors, Arm core or like microcontroller-based devices, CCTV cameras, Baby Monitors, Home Lighting Systems, and the high computation and memory requirements for traditional cryptographic authentication and encryption solutions, there is a need to develop lightweight fully optimized cryptographic security protocols for IoT devices [201]. Application specific functionality vis-a-vis low manufacturing cost and low energy consumption are also considered to be the limiting factors in developing a generalized solution for all the IoT products.Correspondingly, there is a requirement of an international IoT standards enforcing body to enforce minimum security standards in IoT products.

B. Privacy-Preserving Data Aggregation and Processing

Privacy is a critical security requirement for IoT users. Although considerable research has already been done concerning user as well as data privacy, however, certain issues like privacy in data collection, data aggregation, data sharing, and data management warrant further attention [23]. E.g., data aggregation is done at the gateway devices to reduce the communication overhead between end devices and the cloud/ servers. To preserve data security and privacy, the aggregation or processing is done over encrypted data by employing additive [202, 203] or multiplicative homomorphic encryption schemes. There are some full homomorphic encryption schemes as well [204, 205], however, due to heavy computation load, it is difficult to use full homomorphic encryption schemes in IoT. Apart from data encryption, users' signatures aggregation is another approach to contain the communication overhead, given p signatures on p distinct messages from the same user. However, it is quite challenging to design a multi-key homomorphic signature to aggregate p signatures on p distinct messages generated by p users [103].

C. Software/Code Integrity

Numerous solutions to ensure the integrity of IoT end devices exist. However, the most dependable solutions are hardware-based that require execution of complete attestation process in a secure environment. But keeping in view the scale of deployment and low cost of IoT devices, manufacturing of secure hardware-based IoT products for usages besides critical infrastructure is not a practical one. Hence, there is a need to explore a secure software-based solution that can be easily deployed in resource constraint IoT devices with the flexibility of timely upgradation. Another foreseeable problem is that next generation of IoT will consist of a large number of heterogeneous devices. Therefore, to detect and correct any malicious software modification efficiently, a swarm attestation mechanism for large dynamic and heterogeneous networks of embedded systems is still a challenging task [206].

D. Blockchain - An Instrument to Augment IoT Security

The success of Bitcoin brought the attention of the world to its underlying Blockchain technology [162]. The Blockchain is considered to be an unforgeable digital ledger that cannot be manipulated and changed. Although Blockchain was initially developed for fintech (financial technology), yet it is being adopted by many to

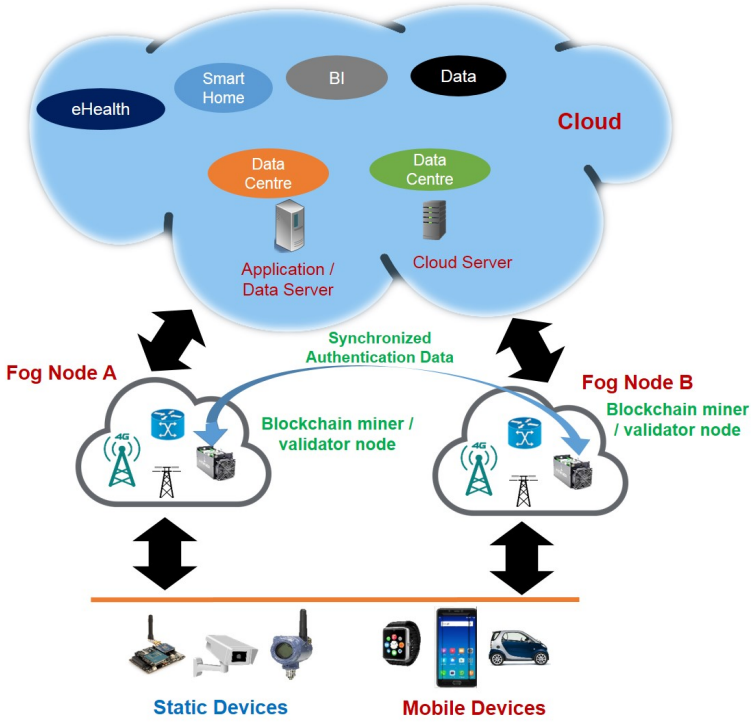

Fig. 17: Blockchain-based ID Authentication in Fog Computing

provide secure distributed services, such as Hyperledger-Fabric by Linux Foundation [165], smart city security [207], supply chain management [208], data sharing [209], data security [210] and decentralized and distributed web services [211]. However, Blockchain's adaptation in IoT ecosystem requires further evaluation. Figure-16 shows the inherent benefits of Blockchain in blue blocks, its limitations in pink blocks and the Blockchain features that can leverage IoT security in green blocks. The open research issues are shown in yellow blocks.

Although IoT can inherit some of the core benefits of Blockchain such as decentralized and unforgeable digital ledger, transaction integrity and authentication, no double spending, trustless operation and by design protection against ransomware and cryptlocker type attacks. However, to make Blockchain a reliable and secure platform for IoT, certain aspects need further research and evaluation. Such challenges include, identity management with due consideration for user privacy, user data privacy (both, on chain and in transit), minimum latency in transaction confirmation for near real-time IoT systems (smart vehicles, autonomous traffic management, smart grid, health monitoring), IoT focused transaction validation rules, IoT centric consensus mechanism with low energy, low computation and low communication overhead. The research on IoT-centric consensus mechanism must focus on consensus finality and fork prevention, which is a key to minimize latency in transaction confirmation and a critical requirement for real-time IoT systems.

E. Challenges to Fog Computing in IoT

One of the challenges in fog computing is to realize identity authentication while ensuring low latency of real-time services, the mobility of users, decentralized fog computing nodes and avoiding de-anonymization attacks [212]. Currently, there are many identity authentication schemes [213, 214, 215]. However, they do not cater for the mobility of the end devices. The probable solution to this challenge lies in the Blockchain-based access control for the fog computing. As shown in Figure-17, all the fog computing nodes can be the full nodes for the Blockchain and can securely share and maintain the users' authentication and authorization information using group keys or attribute-based encryption [216, 217].

Another challenge is the consistency of the access control policy when multiple devices are used by the users to access real-time services. The policy may involve device authentication and management mechanism for the users and key management mechanism for the fog nodes.

Although security is an essential part of any IoT system, however, the limited computational and power capability of IoT devices, makes it difficult to employ conventional cryptographic solutions. Hence, there is a requirement to design lightweight security protocols to support real-time services for fog assisted IoT applications.

VII. Conclusion and Future Work

In this paper, we have tried to highlight most of the known threats to IoT systems by quoting examples of successful attacks. These threats range from simple message interception to sophisticated malware attacks. We have also presented a comprehensive attack methodology for most common real world attacks. We also deduced an attack strategy of a DDoS attack through IoT botnet followed by requisite security measures. This paper also presented a comprehensive set of security guidelines based on industry best practices that can help IoT standardization bodies to design minimum security standards based on types of IoT applications and devices. Finally, some open research challenges related to IoT security were discussed. As for today, the inherent security provided by the communication protocols does not protect against malware and node compromise attacks. Moreover, in the backdrop of a recent upsurge in the number of Ransomware Attacks, the leading cause of their detrimental effects can be attributed to centralized network architecture, in which all the network functionalities and security operations are controlled centrally. Such architectures are costly to set up, and on the other hand, present a single point of failure.

Hence, apart from other techniques, Blockchain technology with its inherent cryptographic security and unforgeable distributed architecture is also being evaluated and tested to address the security and privacy issues of IoT. It is believed that Blockchain can solve most of the data integrity issues of IoT due to its ability to run distributed apps in the form of smart contracts and storing data on multiple nodes. Therefore, we desire to develop a secure Blockchain-based IoT protocol in future, that would aim to protect the IoT systems against most of the integrity attacks.

References

[1] N. Cam-Winget, A.-R. Sadeghi, and Y. Jin, "Can IoT be secured: Emerging challenges in connecting the unconnected," in *Proc. 53rd ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2016, pp. 1–6.

[2] J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, and A. Marrs, "Disruptive technologies: Advances that will transform life, business, and the global economy ," *McKinsey Global Institute San Francisco, CA*, vol. 12, 2013.

[3] D. Lund, C. MacGillivray, V. Turner, and M. Morales, "Worldwide and regional Internet of Things (IoT) 2014–2020 forecast: A virtuous circle of proven value and demand," *International Data Corporation (IDC), Tech. Rep*, 2014.

[4] D. Evans, "The Internet of Things: How the next evolution of the internet is changing everything," *CISCO white paper*, vol. 1, no. 2011, pp. 1–11.

[5] Computer Virus Strikes CSX Transportation Computers. (2003). [Online]. Available: http://www.prnewswire.com/news-releases/computer-virus-strikes-csx-transportation-computers-70971537.html

[6] K. Poulsen, "Slammer worm crashed Ohio nuke plant network," *Security Focus*, vol. 19, 2003.

[7] A. Greenberg, "Hackers remotely kill a jeep on the highwayWith me in it," *Wired*, vol. 7, p. 21, 2015.

[8] S. A. Kumar, T. Vealey, and H. Srivastava, "Security in internet of things: Challenges, solutions and future directions," in *Proc. 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2016, pp. 5772–5781.

[9] A.-R. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial Internet of Things," in *Proc. 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015, pp. 1–6.

[10] T. Borgohain, U. Kumar, and S. Sanyal, "Survey of security and privacy issues of Internet of Things," 2015.

[11] J. Singh, T. Pasquier, J. Bacon, H. Ko, and D. Eyers, "Twenty Security Considerations for Cloud-Supported Internet of Things," *IEEE Internet of Things Journal*, vol. 3, pp. 269–284, 2015.

[12] M. Andrew. How the Internet of Things will affect security & privacy, (2016). [Online]. Available: http://www.businessinsider.com/internet-of-things-security-privacy-2016-8?IR=T

[13] J. Steinberg. These Devices May Be Spying On You (Even In Your Own Home, (2014)). [Online]. Available: https://www.forbes.com/sites/josephsteinberg/2014/01/27/these-devices-may-be-spying-on-you-even-in-your-own-home/#73cc4556b859

[14] Internet of Things Security Study: Smart Watches, (2017). [Online]. Available: http://go.saas.hpe.com/fod/internet-of-things

[15] IBM Point of View: Internet of Things Security, (2015). [Online]. Available: http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=RAW14382USEN

[16] D. Storm. SCADA Strangelove: Zero-days & hacking for full remote control, (2015). [Online]. Available: http://www.computerworld.com/article/2475789/cybercrime-hacking/hackers-exploit-scada-holes-to-take-full-control-of-critical-infrastructure.html

[17] The CEO's Guide to Data Security. Protect your data through innovation - AT&T Cybersecurity Insights (Vol 5), (2016). [Online]. Available: https://www.business.att.com/cybersecurity/docs/vol5-datasecurity.pdf

[18] P. Ducklin. Mirai Internet of Things malware from Krebs DDoS attack goes open source, (2016). [Online]. Available: https://nakedsecurity.sophos.com/2016/10/05/mirai/

[19] R. Brewer, "Ransomware attacks: detection, prevention and cure," *Network Security*, vol. 2016, no. 9, pp. 5–9, 2016.

[20] E. Kovacs. Shamoon Attacks Possibly Aided by Greenbug Group, (2017). [Online]. Available: http://www.securityweek.com/shamoon-attacks-possibly-aided-greenbug-group

[21] Duqu2.0: The Most Sophisticated Malware Ever Seen, (2015). [Online]. Available: http://resources.infosecinstitute.com/duqu-2-0-the-most-sophisticated-malware-ever-seen/\#gref

[22] M. Ahlmeyer and A. M. Chircu, "Securing The Internet of Things: A Review," *Issues in Information Systems*, vol. 17, no. 4, 2016.

[23] M. Abomhara and G. M. Køien, "Security and privacy in the internet of things: Current status and open issues," in *Proc. IEEE International Conference on Privacy and Security in Mobile Systems (PRISMS)*, 2014, pp. 1–8.

[24] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the internet of things: a survey of existing protocols and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.

[25] V. Sivaraman, H. H. Gharakheili, A. Vishwanath, R. Boreli, and O. Mehani, "Network-level security and privacy control for smart-home iot devices," in *Proc. 11th IEEE International Conference onWireless and Mobile Computing, Networking and Communications (WiMob)*, 2015, pp. 163–167.

[26] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu, "Security of the internet of things: perspectives and challenges," *Wireless Networks*, vol. 20, no. 8, pp. 2481–2501, 2014.

[27] A. Banafa, "IoT Standardization and Implementation Challenges," *IEEE Internet of Things Journal*, 2016. [Online]. Available: http://iot.ieee.org/newsletter/july-2016/iot-standardization-and-implementation-challenges.html

[28] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[29] M. Khari, M. Kumar, S. Vij, P. Pandey, and Vaishali, "Internet of Things: Proposed security aspects for digitizing the world," in *Proc. 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 2165–2170.

[30] A. Reziouk, E. Laurent, and J.-C. Demay, "Practical security overview of IEEE 802.15.4," in *Proc. IEEE International Conference on Engineering & MIS (ICEMIS)*, 2016, pp. 1–9.

[31] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[32] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, "Future Internet: The internet of things architecture, possible applications and key challenges," in *Proc. 10th IEEE International Conference on Frontiers of Information Technology (FIT)*, 2012, pp. 257–260.

[33] Z. Yang, Y. Yue, Y. Yang, Y. Peng, X. Wang, and W. Liu, "Study and application on the architecture and key technologies for IoT," in *Proc. IEEE International Conference on Multimedia Technology (ICMT)*, 2011, pp. 747–751.

[34] M. Wu, T.-J. Lu, F.-Y. Ling, J. Sun, and H.-Y. Du, "Research on the architecture of Internet of Things," in *Proc. 3rd IEEE International Conference on Advanced Computer Theory and Engineering (ICACTE)*, vol. 5, 2010, pp. V5–484.

[35] L. Tan and N. Wang, "Future internet: The Internet of Things," in *Proc. 3rd IEEE International Conference on Advanced Computer Theory and Engineering (ICACTE)*, vol. 5, 2010, pp. V5–376.

[36] M. A. Chaqfeh and N. Mohamed, "Challenges in middleware solutions for the Internet of Things," in *Proc. IEEE International Conference on Collaboration Technologies and Systems (CTS)*, 2012, pp. 21–26.

[37] D. Uckelmann, M. Harrison, and F. Michahelles, "An architectural approach towards the future Internet of Things," in *Architecting the internet of things*. Springer, 2011, pp. 1–24.

[38] T. Yu, V. Sekar, S. Seshan, Y. Agarwal, and C. Xu, "Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the internet-of-things," in *Proc. 14th ACM Workshop on Hot Topics in Networks*, 2015, p. 5.

[39] HPE Fortify and the Internet of Things, (2017) . [Online]. Available: http://go.saas.hpe.com/fod/internet-of-things

[40] H. Ning, H. Liu, and L. T. Yang, "Cyberentity security in the internet of things," *Computer*, vol. 46, no. 4, pp. 46–53, 2013.

[41] R. M. Savola, H. Abie, and M. Sihvonen, "Towards metrics-driven adaptive security management in e-health iot applications," in *Proc. 7th International Conference on Body Area Networks*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2012, pp. 276–281.

[42] M. Rostami, F. Koushanfar, and R. Karri, "A primer on hardware security: Models, methods, and metrics," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1283–1295, 2014.

[43] IoT Security: An IBM Position Paper, (2016). [Online]. Available: https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=WWW12379USEN\&

[44] Mirai Internet of Things malware from Krebs DDoS attack goes open source, (2016). [Online]. Available: https://nakedsecurity.sophos.com/2016/10/05/mirai-internet-of-things-malware

[45] A. Burg, A. Chattopadhyay, and K.-Y. Lam, "Wireless communication and security issues for cyber–physical systems and the internet-of-things," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 38–60, 2018.

[46] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, "A survey on wireless security: Technical challenges, recent advances, and future trends," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1727–1765, 2016.

[47] M. Guri, G. Kedma, A. Kachlon, and Y. Elovici, "Airhopper: Bridging the air-gap between isolated networks and mobile phones using radio frequencies," in *Proc. 9th IEEE International Conference on Malicious and Unwanted Software: The Americas (MALWARE)*, 2014, pp. 58–67.

[48] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel, "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *Proc. IEEE Symposium on Security and Privacy*, 2008, pp. 129–142.

[49] P. Schneider and G. Horn, "Towards 5g security," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, vol. 1, 2015, pp. 1165–1170.

[50] R. S. Sinha, Y. Wei, and S.-H. Hwang, "A survey on lpwa technology: Lora and nb-iot," *Ict Express*, vol. 3, no. 1, pp. 14–21, 2017.

[51] M. Chen, Y. Miao, Y. Hao, and K. Hwang, "Narrow band internet of things," *IEEE Access*, vol. 5, pp. 20 557–20 577, 2017.

[52] F. Koushanfar, A.-R. Sadeghi, and H. Seudie, "EDA for secure and dependable cybercars: challenges and opportunities," in *Proc. 49th ACM Annual Design Automation Conference*, 2012, pp. 220–228.

[53] D. L. Lough, "A taxonomy of computer attacks with applications to wireless networks," Ph.D. dissertation, Virginia Tech, 2001.

[54] M. Vanhoef and F. Piessens, "Advanced wi-fi attacks using commodity hardware," in *Proc. 30th ACM Annual Computer Security Applications Conference*, 2014, pp. 256–265.

[55] A. Nasrallah, A. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. ElBakoury, "Ultra-low latency (ull) networks: A comprehensive survey covering the ieee tsn standard and related ull research," *arXiv preprint arXiv:1803.07673*, 2018.

[56] T. Mizrahi, E. Grossman, A. J. Hacker, S. Das, J. Dowdell, H. Austad, K. Stanton, and N. Finn, "Deterministic Networking (DetNet) Security Considerations," Internet Engineering Task Force, Internet-Draft draft-ietf-detnet-security-02, 2018, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-detnet-security-02

[57] M. Furdek, N. Skorin-Kapov, S. Zsigmond, and L. Wosinska, "Vulnerabilities and security issues in optical networks," in *Proc. 16th IEEE International Conference on Transparent Optical Networks (ICTON)*, 2014, pp. 1–4.

[58] B. Everett, "Tapping into fibre optic cables," *Network Security*, vol. 2007, no. 5, pp. 13–16, 2007.

[59] Alcatel-Lucent 1830 Photonic Service Switch (PSS-64 and PSS-36), Alcatel Lucent, (2014). [Online]. Available: http://lightspeedt.com/wp-content/uploads/2015/10/1830-PSS-Datasheet.pdf

[60] C. Mas, I. Tomkos, and O. K. Tonguz, "Failure location algorithm for transparent optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 8, pp. 1508–1519, 2005.

[61] A. Bononi, P. Serena, N. Rossi, and D. Sperti, "Which is the dominant nonlinearity in long-haul pdm-qpsk coherent transmissions?" in *Proc. 36th IEEE European Conference and Exhibition on Optical Communication (ECOC)*, 2010, pp. 1–3.

[62] R. Aparicio-Pardo, P. Pavon-Marino, and S. Zsigmond, "Mixed line rate virtual topology design considering nonlinear interferences between amplitude and phase modulated channels," *Photonic Network Communications*, vol. 22, no. 3, pp. 230–239, 2011.

[63] G. Hernandez, O. Arias, D. Buentello, and Y. Jin, "Smart nest thermostat: A smart spy in your home," *Black Hat USA*, 2014.

[64] S. Zonouz, J. Rrushi, and S. McLaughlin, "Detecting industrial control malware using automated PLC code analytics," *IEEE Security & Privacy*, vol. 12, no. 6, pp. 40–47, 2014.

[65] Attack of Things, (2016). [Online]. Available: http://news.level3.com/2016-08-29-Attack-of-Things

[66] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 2, 2009.

[67] F. B. Thomas. It's Depressingly Easy To Spy On Vulnerable Baby Monitors Using Just A Browser, (2015). [Online]. Available: https://www.forbes.com/sites/thomasbrewster/2015/09/02/baby-surveillance-with-a-browser/\#2508d85b1aa0

[68] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "Threats to networking cloud and edge datacenters in the Internet of Things," *IEEE Cloud Computing*, vol. 3, no. 3, pp. 64–71, 2016.

[69] J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis & defenses," in *Proc. 3rd ACM International Symposium on Information processing in sensor networks*, 2004, pp. 259–268.

[70] F.-X. Standaert, "Introduction to side-channel attacks," in *Secure Integrated Circuits and Systems*. Springer, 2010, pp. 27–42.

[71] J. Wurm, K. Hoang, O. Arias, A.-R. Sadeghi, and Y. Jin, "Security analysis on consumer and industrial IoT devices," in *Proc. 21st IEEE Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2016, pp. 519–524.

[72] O. Arias, J. Wurm, K. Hoang, and Y. Jin, "Privacy and security in internet of things and wearable devices," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 1, no. 2, pp. 99–109, 2015.

[73] Xbox 360 Timing Attack, (2007). [Online]. Available: {http://beta.ivc.no/wiki/index.php/Xbox_360_Timing_Attack}

[74] B. Balamurugan and B. Dyutimoy, "Security in network layer of iot: Possible measures to preclude," in *Security Breaches and Threat Prevention in the Internet of Things*, J. N. and T. R., Eds. IGI Global, 2017, ch. 3, pp. 46–75.

[75] S. Skorobogatov, "Fault attacks on secure chips: from glitch to flash," *Design and Security of Cryptographic Algorithms and Devices (ECRYPT II)*, 2011.

[76] R. Lemos, "Sony left passwords, code-signing keys virtually unprotected," *eWeek*, 2014.

[77] B. Fowler, "Some top baby monitors lack basic security features report finds," 2015.

[78] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "A dynamic prime number based efficient security mechanism for big sensing data streams," *Journal of Computer and System Sciences*, vol. 83, no. 1, pp. 22–42, 2017.

[79] S. Mohammadi and H. Jadidoleslamy, "A comparison of link layer attacks on wireless sensor networks," *arXiv preprint arXiv:1103.5589*, 2011.

[80] V. B. Misic, J. Fang, and J. Misic, "MAC layer security of 802.15.4-compliant networks," in *Proc. IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, 2005, pp. 8–pp.

[81] N. Sastry and D. Wagner, "Security considerations for IEEE 802.15. 4 networks," in *Proc. Proceedings of the 3rd ACM workshop on Wireless security*, 2004, pp. 32–42.

[82] R. Riaz, K.-H. Kim, and H. F. Ahmed, "Security analysis survey and framework design for ip connected lowpans," in *Proc. IEEE International Symposium on Autonomous Decentralized Systems, ISADS'09.*, 2009, pp. 1–6.

[83] D. R. Raymond and S. F. Midkiff, "Denial-of-service in wireless sensor networks: Attacks and defenses," *IEEE Pervasive Computing*, vol. 7, no. 1, 2008.

[84] Y. Sun, L. Wu, S. Wu, S. Li, T. Zhang, L. Zhang, J. Xu, Y. Xiong, and X. Cui, "Attacks and countermeasures in the internet of vehicles," *Annals of Telecommunications*, vol. 72, no. 5, pp. 283–295, Jun 2017. [Online]. Available: https://doi.org/10.1007/s12243-016-0551-6

[85] F. Shahzad, M. Pasha, and A. Ahmad, "A survey of active attacks on wireless sensor networks and their countermeasures," *CoRR*, vol. abs/1702.07136, 2017. [Online]. Available: http://arxiv.org/abs/1702.07136

[86] J. Murphy, "Enhanced Security Controls for IBM Watson IoT Platform," *IBM Watson IoT Platform*, 2016. [Online]. Available: https://developer.ibm.com/iotplatform/2016/09/23/enhanced-security-controls-for-ibm-watson-iot-platform/

[87] Secure Adaptive Routing Protocol for Wireless Sensor Networks, (2018). [Online]. Available: {https://www.dfcsc.uri.edu/docs/posters/sarp.pdf}

[88] A. Kanuparthi, R. Karri, and S. Addepalli, "Hardware and embedded security in the context of internet of things," in *Proc. ACM workshop on Security, privacy & dependability for cyber vehicles*, 2013, pp. 61–64.

[89] OWASP Top 10 2017 - The Ten Most Critical Web Application Security Risks, (2017). [Online]. Available: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_2017_Project

[90] SQLi, XSS zero-days expose Belkin IoT devices, Android smartphones, (2016). [Online]. Available: https://www.csoonline.com/article/3138935/security/sqli-xss-zero-days-expose-belkin-iot-devices-android-smartphones.html

[91] Cross-site Scripting (XSS) Attack, (2018). [Online]. Available: https://www.acunetix.com/websitesecurity/cross-site-scripting/

[92] K. Hamlen, M. Kantarcioglu, L. Khan, and B. Thuraisingham, "Security issues for cloud computing," *Optimizing Information Security and Advancing Privacy Assurance: New Technologies*, vol. 150, 2012.

[93] What is Cloud Computing, (2018). [Online]. Available: https://www.ibm.com/cloud/learn/what-is-cloud-computing

[94] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 50–55, 2008.

[95] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of internet services and applications*, vol. 1, no. 1, pp. 7–18, 2010.

[96] A. Oliner, A. Ganapathi, and W. Xu, "Advances and challenges in log analysis," *Communications of the ACM*, vol. 55, no. 2, pp. 55–61, 2012.

[97] S. Yu, *Distributed Denial of Service Attack and Defense*. Springer, 2014.

[98] T. Moore, J. Clulow, S. Nagaraja, and R. Anderson, "New strategies for revocation in ad-hoc networks," in *Proc. European Workshop on Security in Ad-hoc and Sensor Networks*. Springer, 2007, pp. 232–246.

[99] Ransomware Holding Your Data Hostage, Deloitte: Threat Intelligence and Analytics, (2016). [Online]. Available: https://www2.deloitte.com/content/dam/Deloitte/us/Documents/risk/us-aers-ransomware.pdf

[100] Cisco, "Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are," *Cisco Whitepaper*, 2015.

[101] L. M. Vaquero and L. Rodero-Merino, "Finding your way in the fog: Towards a comprehensive definition of fog computing," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 27–32, 2014.

[102] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A comprehensive survey on fog computing: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 416–464, 2017.

[103] J. Ni, K. Zhang, X. Lin, and X. Shen, "Securing fog computing for internet of things applications: Challenges and solutions," *IEEE Communications Surveys & Tutorials*, 2017.

[104] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems*, 2016.

[105] K. Zhang, X. Liang, R. Lu, K. Yang, and X. S. Shen, "Exploiting mobile social behaviors for sybil detection," in *Proc. IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 271–279.

[106] S. Bruce. Schneier on Security: The Future of Ransomware, (2017). [Online]. Available: https://www.schneier.com/blog/archives/2017/05/the_future_of_r.html

[107] F. B. Lorenzo. Hackers Make the First-Ever Ransomware for Smart Thermostats, (2016). [Online]. Available: https://motherboard.vice.com/en_us/article/aekj9j/internet-of-things-ransomware-smart-thermostat

[108] History of Viruses, NIST Computer Security Resource Center, (1994). [Online]. Available: {http://csrc.nist.gov/publications/nistir/threats/subsubsection3_3_1_1.html}

[109] Timeline of computer viruses and worms, (2017). [Online]. Available: {https://en.wikipedia.org/wiki/Timeline_of_computer_viruses_and_worms}

[110] National Cyber Awareness System, US CERT, (2018) . [Online]. Available: https://www.us-cert.gov/ncas/alerts

[111] New Petya / NotPetya / ExPetr ransomware outbreak, Kaspersky Lab, (2017). [Online]. Available: https://blog.kaspersky.com/new-ransomware-epidemics/17314/

[112] Duqu 2.0: The Most Sophisticated Malware Ever Seen, (2015). [Online]. Available: http://resources.infosecinstitute.com/duqu-2-0-the-most-sophisticated-malware-ever-seen/\#gref

[113] Microsoft Fixes 3 Zero-day and many other flaws being exploited in the Wild, (2014). [Online]. Available: http://securityaffairs.co/wordpress/29270/security/microsoft-fixes-3-zero-day.html

[114] Microsoft issued a critical Out-of-Band patch for Kerberos flaw, (2014). [Online]. Available: http://securityaffairs.co/wordpress/30320/security/microsoft-patch-kerberos-bug.html

[115] What exactly is Duqu 2.0?, Rapid7 Community, (2015). [Online]. Available: https://community.rapid7.com/community/infosec/blog/2015/06/12/what-exactly-is-duqu-20

[116] R. Brewer, "Ransomware attacks: detection, prevention and cure," *Network Security*, vol. 2016, no. 9, pp. 5–9, 2016.

[117] C. Bronk and E. Tikk-Ringas, "The cyber attack on Saudi Aramco," *Survival*, vol. 55, no. 2, pp. 81–96, 2013.

[118] S. Zhioua, "The Middle East under Malware Attack Dissecting Cyber Weapons," in *Proc. 33rd IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW)*, 2013, pp. 11–16.

[119] Shamoon Attacks Possibly Aided by Greenbug Group, (2017). [Online]. Available: http://www.securityweek.com/shamoon-attacks-possibly-aided-greenbug-group

[120] Shamoon return prompts Saudi Arabia cyber warning, (2017). [Online]. Available: http://www.smh.com.au/world/shamoon-return-prompts-saudi-arabia-cyber-warning-20170124-gtxggi.html

[121] B. Miller and D. Rowe, "A survey SCADA of and critical infrastructure incidents," in *Proc. 1st ACM Annual conference on Research in information technology*, 2012, pp. 51–56.

[122] E. Nakashima, G. Miller, and J. Tate, "US, Israel developed Flame computer virus to slow Iranian nuclear efforts, officials say," The Washington Post, 2012.

[123] B. Bencsáth, G. Pék, L. Buttyán, and M. Felegyhazi, "The cousins of stuxnet: Duqu, flame, and gauss," *Future Internet*, vol. 4, no. 4, pp. 971–1003, 2012.

[124] A. Gostev, "The flame: Questions and answers," *Securelist,{Online resource} Available at: https://www. securelist. com/en/blog/208193522/The_Flame_Questions_and_Answers*, 2012.

[125] Common Vulnerabilities and Exposures-CVE-2010-2568, (2010). [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2010-2568

[126] GReAT. Gauss: Nation-state Cyber-surveillance Meets Banking Trojan; Technical report; Kaspersky Labs: Moscow, Russian, Kaspersky Labs, (2012) . [Online]. Available: https://securelist.com/gauss-nation-state-cyber-surveillance-meets-banking-trojan-54/33854/

[127] The Icefog APT: A Tale of Cloak and Three Daggers, Kaspersky Labs, (2013). [Online]. Available: https://securelist.com/the-icefog-apt-a-tale-of-cloak-and-three-daggers/57331/

[128] Russian-Based Dragonfly Group Attacks Energy Industry, RISI Online Incident database, (2015). [Online]. Available: http://www.risidata.com/Database/event_date/desc

[129] Dragonfly: Western Energy Companies Under Sabotage Threat. Symantec Security Response, Symantect, (2016). [Online]. Available: https://www.symantec.com/connect/blogs/dragonfly-western-energy-companies-under-sabotage-threat

[130] GReAT. Red October - Diplomatic Cyber Attacks Investigation, Kaspersky Labs (2014). [Online]. Available: https://securelist.com/red-october-diplomatic-cyber-attacks-investigation/36740/

[131] G. Wangen, "The role of malware in reported cyber espionage: a review of the impact and mechanism," *Information*, vol. 6, no. 2, pp. 183–211, 2015.

[132] N. Virvilis and D. Gritzalis, "The big four-what we did wrong in advanced persistent threat detection?" in *Proc. Eighth IEEE International Conference on Availability, Reliability and Security (ARES)*, 2013, pp. 248–254.

[133] Night Dragon Attacks Target Technology in Energy Sector, Forbes, (2011). [Online]. Available: http://www.forbes.com/sites/williampentland/2011/02/19/night-dragon-attacks-target-technology-in-energy-industry/\#28c010114301

[134] Trojans exploit WAP subscriptions to steal money, Kaspersky Lab, (2017). [Online]. Available: https://www.kaspersky.com/blog/wap-billing-trojans/18080/

[135] J. M. Ehrenfeld, "WannaCry, Cybersecurity and Health Information Technology: A Time to Act," *Journal of Medical Systems*, vol. 41, no. 7, p. 104, 2017. [Online]. Available: http://dx.doi.org/10.1007/s10916-017-0752-1

[136] W. Victoria. WannaCry Ransomware: what is it and how to protect yourself, (2017). [Online]. Available: http://www.wired.co.uk/article/wannacry-ransomware-virus-patch

[137] C. Roger. NHS ransomware attack spreads worldwide, (2017). [Online]. Available: http://www.cmaj.ca/content/189/22/E786

[138] R. Carol. The impact of WannaCry on industrial control systems (ICS), (2016). [Online]. Available: http://iiot-world.com/cybersecurity/the-impact-of-wannacry-on-industrial-control-systems-ics/

[139] D. Goodin, "Youre infected: If you want to see your data again, pay US $300 in Bitcoins," *Ars Technica*, 2013.

[140] D. Oberhaus. This Luxury Hotel Is Sick of Ransomware Attacks, So It's Going Analog, (2017). [Online]. Available: {https://motherboard.vice.com/en_us/article/nzdznb/luxury-hotel-goes-analog-to-fight-ransomware-attacks}

[141] Mirai: what you need to know about the botnet behind recent major DDoS attacks, Symantec, (2016). [Online]. Available: https://www.symantec.com/connect/blogs/mirai-what-you-need-know-about-botnet-behind-recent-major-ddos-attacks

[142] ICS-ALERT-14-176-02A, ICS-CERT, (2014) . [Online]. Available: https://ics-cert.us-cert.gov/alerts/ICS-ALERT-14-176-02A

[143] Havex Hunts For ICS/SCADA Systems, (2014). [Online]. Available: https://www.f-secure.com/weblog/archives/00002718.html

[144] N. Falliere, L. O. Murchu, and E. Chien, "W32. stuxnet dossier," *White paper, Symantec Corp., Security Response*, vol. 5, no. 6, 2011.

[145] R. Langner, "To kill a centrifuge: A technical analysis of what stuxnets creators tried to achieve," 2013. [Online]. Available: https://www.langner.com/wp-content/uploads/2017/03/to-kill-a-centrifuge.pdf

[146] T. M. Chen and S. Abu-Nimeh, "Lessons from stuxnet," *Computer*, vol. 44, no. 4, pp. 91–93, 2011.

[147] K. Angrishi, "Turning internet of things (iot) into internet of vulnerabilities (iov): Iot botnets," *arXiv preprint arXiv:1702.03681*, 2017.

[148] T. Winter, "RPL: IPv6 routing protocol for low-power and lossy networks," 2012.

[149] E. Rescorla and N. Modadugu, "Datagram transport layer security version 1.2," 2012.

[150] ISO 27001 Risk Assessments, IT Governance U.K., (2017). [Online]. Available: https://www.itgovernance.co.uk/iso27001/iso27001-risk-assessment

[151] NIST, Guide for Conducting Risk Assessment, (2012). [Online]. Available: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf

[152] Do you have a defense-in-depth security strategy?, CISCO, (2017). [Online]. Available: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf

[153] B. Greenstein, "IoT devices used in DDoS Attacks," *IBM Internet of Things Blogs*, 2016. [Online]. Available: https://www.ibm.com/blogs/internet-of-things/ddos-iot-platform-security/

[154] K. Lewis, "IoT security: What are the keys to protecting the castle 247?" *IBM Internet of Things Blogs*, 2017. [Online]. Available: https://www.ibm.com/blogs/internet-of-things/security-iot-ibm/

[155] Guidance for Securing IoT using TCG Technology, Version 1, Revision 21, (2015) . [Online]. Available: https://www.business.att.com/cybersecurity/docs/vol5-datasecurity.pdf

[156] TCG Infrastructure WG TPM Keys for Platform Identity for TPM 1.2, (2015). [Online]. Available: https://trustedcomputinggroup.org/wp-content/uploads/TPM_Keys_for_Platform_Identity_v1_0_r3_Final.pdf

[157] Five Indisputable Facts about IoT Security, IBM Security, (2017). [Online]. Available: https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=SEF03018USEN

[158] ARM mbed, (2017). [Online]. Available: https://www.mbed.com/en/

[159] M. Petko and B. Mark D., "Ima/evm: Real applications for embedded networking systems," in *Proc. Linux Security Summit, Seattle, WA*, 2015.

[160] Secure Authentication and Anti-Counterfeit Technology, (2017). [Online]. Available: http://www.nxp.com/products/identification-and-security/secure-authentication-and-anti-counterfeit-technology:MC_71548

[161] S. Carpov, T. H. Nguyen, R. Sirdey, G. Constantino, and F. Martinelli, "Practical Privacy-Preserving Medical Diagnosis Using Homomorphic Encryption," in *Proc. 9th IEEE International Conference on Cloud Computing (CLOUD)*, June 2016, pp. 593–599.

[162] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[163] B. Smith and K. Christidis, "IBM Blockchain: An Enterprise Deployment of a Distributed Consensus-based Transaction Log," in *Proc. Fourth International IBM Cloud Academy Conference*, 2016, pp. 140–143.

[164] Microsoft Azure, (2017). [Online]. Available: https://azure.microsoft.com/en-au/?&WT.srch=1&WT.mc_ID=AID623263_SEM_MmqDz7Ol

[165] Hyperledger Business Blockchain Technologies, The Linux Foundation, (2017). [Online]. Available: https://www.hyperledger.org/projects

[166] J. Matias, J. Garay, A. Mendiola, N. Toledo, and E. Jacob, "Flownac: Flow-based network access control," in *Proc. Third IEEE European Workshop on Software Defined Networks (EWSDN)*, 2014, pp. 79–84.

[167] F. Jazib, C. Pignataro, A. Jeff, and M. Monique, "Securing the Internet of Things: A Proposed Framework," *Cisco Security Research & Operations*, 2015. [Online]. Available: http://www.cisco.com/c/en/us/about/security-center/secure-iot-proposed-framework.html

[168] S. Chen, M. Ma, and Z. Luo, "An authentication scheme with identity-based cryptography for m2m security in cyber-physical systems," *Security and Communication Networks*, vol. 9, no. 10, pp. 1146–1157, 2016.

[169] Y. Qiu and M. Ma, "A mutual authentication and key establishment scheme for m2m communication in 6lowpan networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2074–2085, 2016.

[170] Y. Qiu, M. Ma, and S. Chen, "An anonymous authentication scheme for multi-domain machine-to-machine communication in cyber-physical systems," *Computer Networks*, vol. 129, pp. 306–318, 2017.

[171] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, "Iot-oas: An oauth-based authorization service architecture for secure services in iot scenarios," *IEEE sensors journal*, vol. 15, no. 2, pp. 1224–1234, 2015.

[172] Information Security Advice: Network Segmentation and Segregation, Australian Government Department of Defence, (2012). [Online]. Available: https://www.asd.gov.au/publications/protect/network_segmentation_segregation.htm

[173] O. Flauzac, C. González, A. Hachani, and F. Nolot, "Sdn based architecture for iot and improvement of the security," in *Proc. 29th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, 2015, pp. 688–693.

[174] A. S. Thyagaturu, A. Mercian, M. P. McGarry, M. Reisslein, and W. Kellerer, "Software defined optical networks (sdons): A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2738–2786, 2016.

[175] Save the Data: Self Encrypting Drives, TCG, (2017). [Online]. Available: https://trustedcomputinggroup.org/wp-content/uploads/Infographic-TCG-SED.pdf

[176] F. Abdi, M. Hasan, S. Mohan, D. Agarwal, and M. Caccamo, "Resecure: A restart-based security protocol for tightly actuated hard real-time systems," *IEEE CERTS*, pp. 47–54, 2016.

[177] A. Meshram and C. Haas, "Anomaly detection in industrial networks using machine learning: a roadmap," in *Machine Learning for Cyber Physical Systems*. Springer, 2017, pp. 65–72.

[178] I. Indre and C. Lemnaru, "Detection and prevention system against cyber attacks and botnet malware for information systems and Internet of Things," in *Proc. 12th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, Sept 2016, pp. 175–182.

[179] S. M. A. M. Gadal and R. A. Mokhtar, "Anomaly detection approach using hybrid algorithm of data mining technique," in *Proc. International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE)*, Jan 2017, pp. 1–6.

[180] B. Mokhtar and M. Azab, "Survey on security issues in vehicular ad hoc networks," *Alexandria Engineering Journal*, vol. 54, no. 4, pp. 1115–1126, 2015.

[181] M. Lee. 2016 Saw An Insane Rise In The Number Of Ransomware Attacks, Forbes, (2016). [Online]. Available: https://www.forbes.com/sites/leemathews/2017/02/07/2016-saw-an-insane-rise-in-the-number-of-ransomware-attacks/#2aad814658dc

[182] Ransomware: 5 Dos and Don'ts, Symantec Corporation, (2016). [Online]. Available: https://us.norton.com/internetsecurity-malware-ransomware-5-dos-and-donts.html

[183] LPWA Technology : Security Comparison, A White paper by Franklin Health Ltd, (2017). [Online]. Available: https://fhcouk.files.wordpress.com/2017/05/lpwa-technology-security-comparison.pdf

[184] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "Threats to networking cloud and edge datacenters in the internet of things," *IEEE Cloud Computing*, vol. 3, no. 3, pp. 64–71, 2016.

[185] N. Kshetri, "Can blockchain strengthen the internet of things?" *IT Professional*, vol. 19, no. 4, pp. 68–72, 2017.

[186] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "Blockchain-based database to ensure data integrity in cloud computing environments," January 2017. [Online]. Available: https://eprints.soton.ac.uk/411996/

[187] S. Sara and N. Michael, "Facebook has been worried about data leaks like this since it went public in 2012,," *CNBC*, 2018. [Online]. Available: https://www.cnbc.com/2018/04/12/facebook-warned-of-data-breaches-years-ago-when-it-went-public-in-2012.html

[188] K. Granville, "Facebook and Cambridge Analytica: What You Need to Know as Fallout Widens," *New York Times*, 2018. [Online]. Available: https://www.nytimes.com/2018/03/19/technology/facebook-cambridge-analytica-explained.html

[189] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, 2017.

[190] A. Miller, Y. Xia, K. Croman, E. Shi, and D. Song, "The honey badger of bft protocols," in *Proc. ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 31–42.

[191] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Communications of the ACM*, vol. 21, no. 7, pp. 558–565, 1978.

[192] F. B. Schneider, "Implementing fault-tolerant services using the state machine approach: A tutorial," *ACM Computing Surveys (CSUR)*, vol. 22, no. 4, pp. 299–319, 1990.

[193] Neo- whitepaper, (2017). [Online]. Available: http://docs.neo.org/en-us/

[194] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "Blockchain-based database to ensure data integrity in cloud computing environments," 2017.

[195] L. Aniello, R. Baldoni, E. Gaetani, F. Lombardi, A. Margheri, and V. Sassone, "A prototype evaluation of a tamper-resistant high performance blockchain-based transaction log for a distributed database," in *Proc. 13th IEEE European Dependable Computing Conference (EDCC)*, 2017, pp. 151–154.

[196] G. Zyskind, O. Nathan, and A. Pentland, "Enigma: Decentralized computation platform with guaranteed privacy," *CoRR*, vol. abs/1506.03471, 2015. [Online]. Available: http://arxiv.org/abs/1506.03471

[197] Y. Li, X. Cheng, Y. Cao, D. Wang, and L. Yang, "Smart choice for the smart grid: Narrowband internet of things (nb-iot)," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1505–1515, 2018.

[198] S. Johan. Firmware Updates over Low-Power Wide Area Networks, The Things Network, (2017). [Online]. Available: https://www.thethingsnetwork.org/article/firmware-updates-over-low-power-wide-area-networks

[199] A. Elsaeidy, I. Elgendi, K. S. Munasinghe, D. Sharma, and A. Jamalipour, "A smart city cyber security platform for narrowband networks," in *Proc. 27th IEEE InternationalTelecommunication Networks and Applications Conference (ITNAC)*, 2017, pp. 1–6.

[200] T. Pecorella, L. Brilli, and L. Mucchi, "The role of physical layer security in iot: A novel perspective," *Information*, vol. 7, no. 3, p. 49, 2016.

[201] O. Vermesan and P. Friess, *Internet of things: converging technologies for smart environments and integrated ecosystems*. River Publishers, 2013.

[202] P. Paillier *et al.*, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. Eurocrypt*, vol. 99. Springer, 1999, pp. 223–238.

[203] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts." in *Proc. TCC*, vol. 3378. Springer, 2005, pp. 325–341.

[204] C. Gentry *et al.*, "Fully homomorphic encryption using ideal lattices." in *Proc. STOC*, vol. 9, no. 2009, 2009, pp. 169–178.

[205] M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Proc. Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2010, pp. 24–43.

[206] N. Asokan, F. Brasser, A. Ibrahim, A.-R. Sadeghi, M. Schunter, G. Tsudik, and C. Wachsmann, "Seda: Scalable embedded device attestation," in *Proc. 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 964–975.

[207] K. Biswas and V. Muthukkumarasamy, "Securing smart cities using blockchain technology," in *Proc. 14th IEEE International Conference on Smart City High Performance Computing and Communications*, 2016, pp. 1392–1393.

[208] W. Reid. How bitcoin's technology could make supply chains more transparent, (2015). [Online]. Available: http://www.coindesk.com/how-bitcoins-technology-could-make-supply-chains/

[209] Implement iot and blockchain for accountability and security, ibm watson iot, (2017). [Online]. Available: https://www.ibm.com/internet-of-things/platform/private-blockchain/

[210] Blockchain startup factom, inc. raises series a funding, (2016). [Online]. Available: https://www.factom.com/news/factom-raises-series-a-funding

[211] D. Lee, "Arachneum: Blockchain meets distributed web," *arXiv preprint arXiv:1609.02789*, 2016.

[212] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *Proc. 30th IEEE Symposium on Security and Privacy*, 2009, pp. 173–187.

[213] V. Odelu, A. K. Das, M. Wazid, and M. Conti, "Provably secure authenticated key agreement scheme for smart grid," *IEEE Transactions on Smart Grid*, 2016.

[214] Q. Jiang, J. Ma, F. Wei, Y. Tian, J. Shen, and Y. Yang, "An untraceable temporal-credential-based two-factor authentication scheme using ecc for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 76, pp. 37–48, 2016.
[215] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang, and X. Yao, "Security and privacy preservation scheme of face identification and resolution framework using fog computing in internet of things," *IEEE Internet of Things Journal*, 2017.
[216] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2011, pp. 568–588.
[217] A. Sahai, B. Waters *et al.*, "Fuzzy identity-based encryption." in *Eurocrypt*, vol. 3494. Springer, 2005, pp. 457–473.

Imran Makhdoom received the B.E (Telecommunications Engineering) and Masters in Information Security degrees from National University of Sciences and Technology, Pakistan in 2004 and 2015, respectively. He has worked as a Project Manager on various wireless communication and IT projects involving Satellite, OFC and CISCO networks. He has also served in a semi-government organization for various cyber-security auditing tasks from 2014 till 2016. He is an EC-Council Certified Secure Computer User (CSCU). Currently, he is a PhD student at University of Technology Sydney and pursuing research on IoT security.

Mehran Abolhasan completed his B.E in Computer Engineering and PhD in Telecommunications on 1999 and 2003 respectively at the University of Wollongong. From 2003-2004, he worked at Smart Internet Technology CRC and Office of Information and Communication Technology within the Department of Commerce in NSW, Australia. In 2004, he joined the Desert knowledge CRC and Telecommunication and IT Research (TITR) Institute to work on a joint project called the Sparse Ad hoc network for Deserts project (Also known as the SAND project). During 2004 to 2007. A/Prof. Abolhasan led a team of researchers at TITR to develop prototype networking devices for rural and remote communication scenarios. Furthermore, he led the deployment of a number of test-beds and field studies in that period. In 2008, he served as a Director of Emerging Networks and Applications Lab (ENAL) at the ICTR institute. During this time, he won a number of major research project grants including an ARC DP project and a number of CRC and other government and industry-based grants. In March 2010, he accepted the position of Senior Lecturer within the faculty of Engineering and IT (FEIT) at the University of Technology Sydney (UTS), where he is now an Associate Professor. In 2014, he accepted the position of Director of Research Programs at FEIT. In 2016, he was appointed as a Deputy Head of School for Research in School of Computing and Communications, UTS. A/Prof. Abolhasan has authored over 100 international publications and has won over one million dollars in research funding. His Current research Interests are in; Software Defined Networking, IoT, Wireless Mesh, Wireless Body Area Networks, 5G Networks and Sensor networks. He is currently a Senior Member of IEEE.

Justin Lipman is an Associate Professor at the University of Technology Sydney focused on research and industry engagement for the Internet of Things, Industrial IoT, Intelligent Transport and Smart Cities. He received his PhD Telecommunications and BE Computer Engineering from the University of Wollongong, Australia in 2003 and 1999 respectively. From 2004 to 2017, he was based in Shanghai, China and held a number of senior management and technical leadership roles at Intel and Alcatel leading research and innovation, product architecture and IP generation. Dr. Lipman has consulted for a number of startups and co-founded two startups. He is an IEEE senior member, with over 40 peer reviewed publications, more than 20 USPTO patents awarded and a further 20 USPTO patent submissions under review. Dr. Lipman is a committee member in Standards Australia contributing to International IoT standards. His research interests are in all things adaptive, connected, distributed and ubiquitous.

Wei Ni received the B.E. and Ph.D. degrees in Electronic Engineering from Fudan University, Shanghai, China, in 2000 and 2005, respectively. Currently, he is a Senior Research Scientist and Team Leader, Data61, CSIRO, Sydney, Australia. Prior to this, he was a Senior Researcher, Devices R&D, Nokia (Jan 2008 March 2009), and a Research Scientist and Deputy Project Manager, Research & Innovation (R&I) Center, Bell Labs, Alcatel/Alcatel-Lucent (Jan 2005 Dec 2007). His efforts led to an Alcatel-Lucent internal venture and three product projects, ten accepted IEEE standard technical proposals, and 25 patents. He has published 38 journal papers and 29 conference papers. Dr Wei Ni serves as an Editor for Hindawi Journal of Engineering since 2012, Secretary of IEEE VTS NSW Chapter since 2014, Track Chair of VTC16-Spring, Track Chair of VTC17-Spring, Chair of Student Travel Grant for WPMC 2014, and Publication Chair of ISCIT 2015.